Using the Domain Name System for System Break-ins

Steven M. Bellovin

Presented by:

Thomas Repantis

`trep@cs.ucr.edu`

Overview

Using DNS to spoof a host’s name and access network services that rely on the host name for authentication.

1. Introduction to the Domain Name System
2. Description of the Attack
3. Proposed Defenses
4. Current Status

Domain Name System (DNS)

- A distributed database, used to map host names to IP addresses, and vice-versa.
- www.cs.ucr.edu
  138.23.169.15
- Paul Mockapetris
  RFCs 882, 883 (1983)
  RFCs 1034, 1035 (1987)

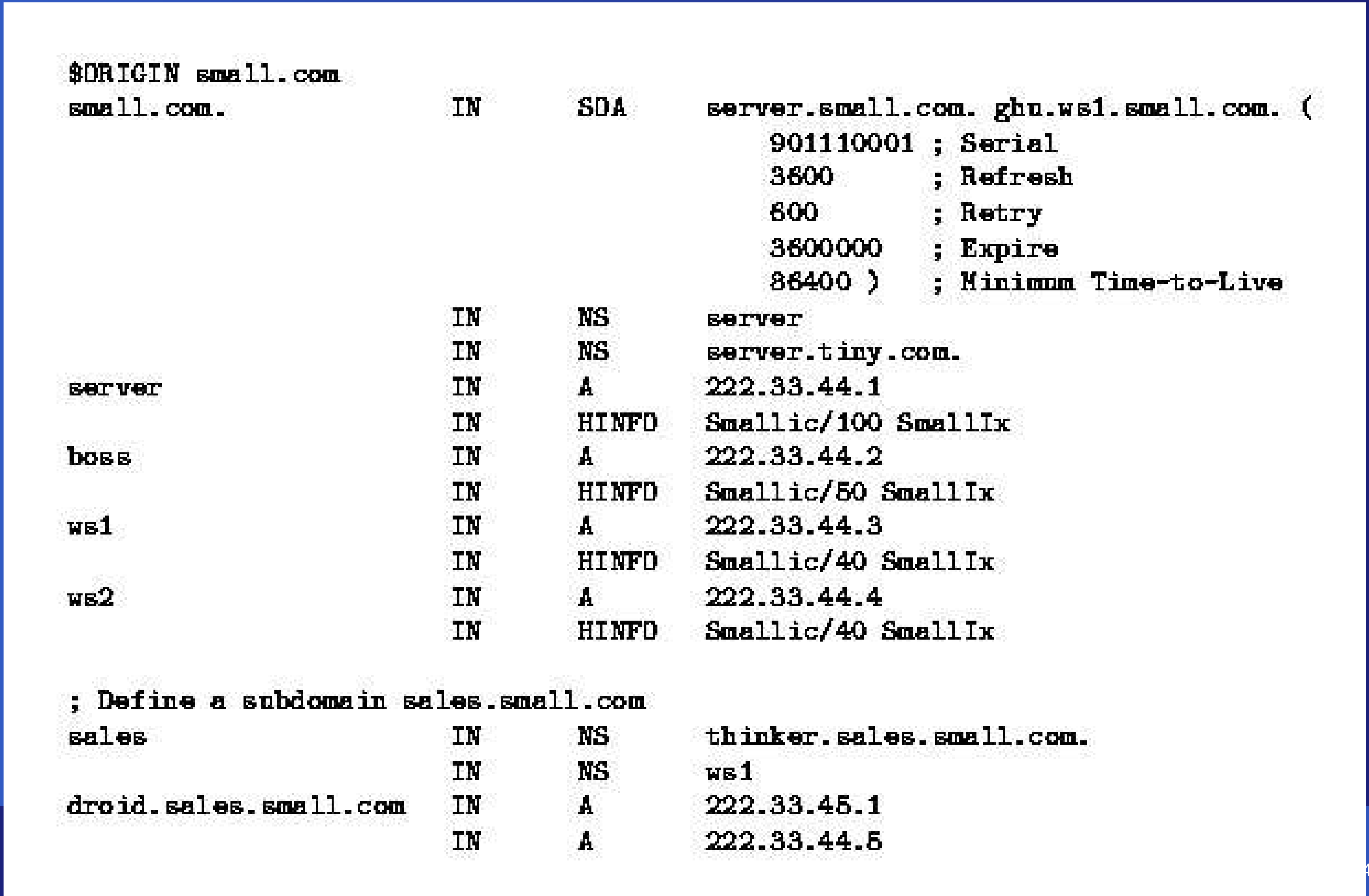
DNS Basics 1/2

- Periods in domain names define zones (www.example.com).
- Servers contain the authoritatitive data for each zone.
- Secondary authoritative servers poll the primary servers.
- If the data has changed, they initiate zone transfers.

DNS Basics 2/2

- The resource records returned are cached locally for some time.
- The authority for a subdomain may be delegated to a subsidiary server (hierarchical namespace).

Zone Example 1/5

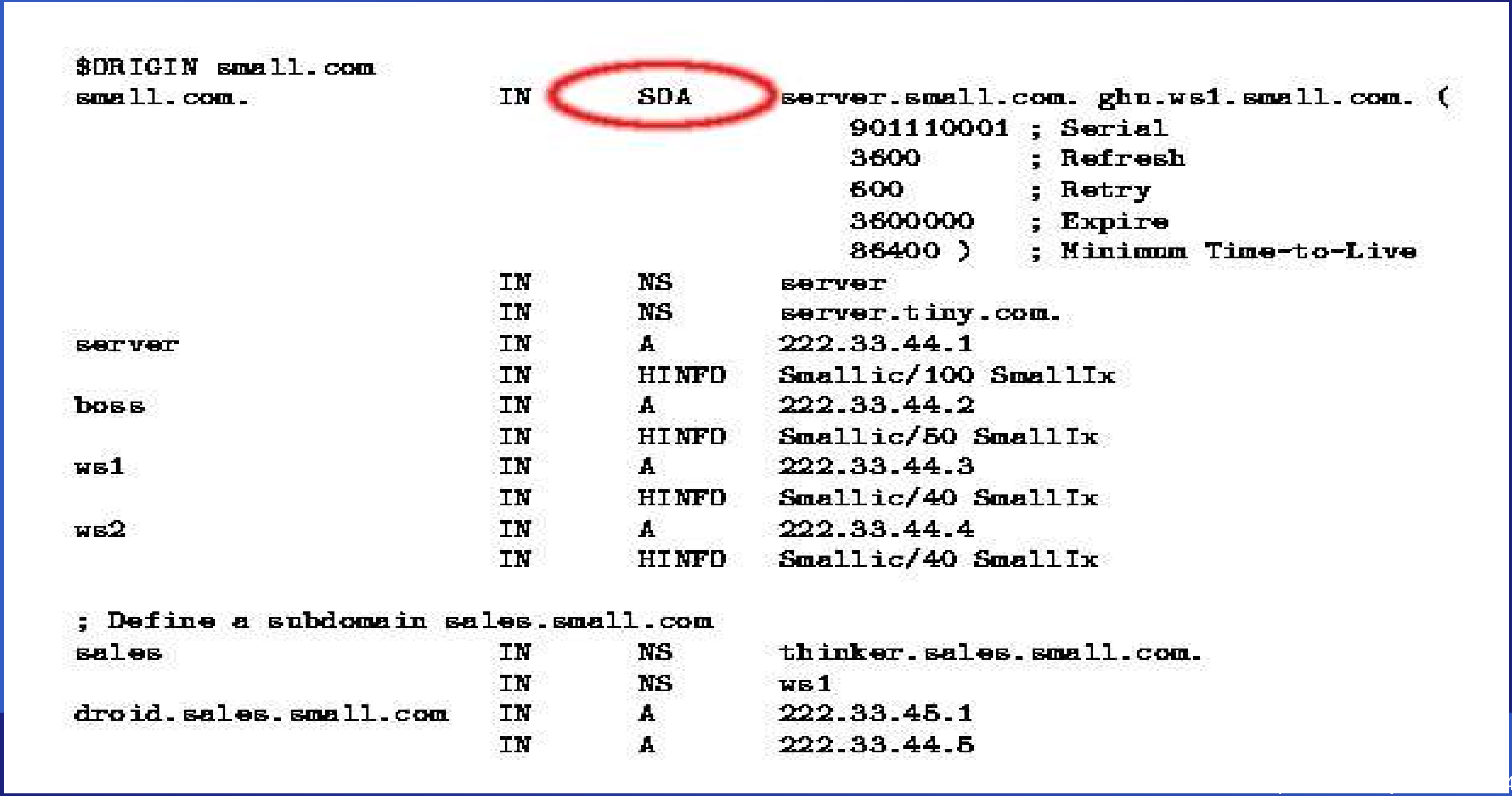


```
$ORIGIN small.com
small.com.          IN    SOA    server.small.com. ghu.ws1.small.com. (
                                     901110001 ; Serial
                                     3600      ; Refresh
                                     600       ; Retry
                                     3600000   ; Expire
                                     86400 )   ; Minimum Time-to-Live
                    IN    NS     server
                    IN    NS     server.tiny.com.
server              IN    A      222.33.44.1
                    IN    HINFO  Smallic/100 SmallIx
boss                IN    A      222.33.44.2
                    IN    HINFO  Smallic/50 SmallIx
ws1                 IN    A      222.33.44.3
                    IN    HINFO  Smallic/40 SmallIx
ws2                 IN    A      222.33.44.4
                    IN    HINFO  Smallic/40 SmallIx

; Define a subdomain sales.small.com
sales               IN    NS     thinker.sales.small.com.
                    IN    NS     ws1
droid.sales.small.com IN  A      222.33.45.1
                    IN    A      222.33.44.5
```

Zone Example 2/5

Start Of Authority (SOA):

Specifies the source of the zone information.

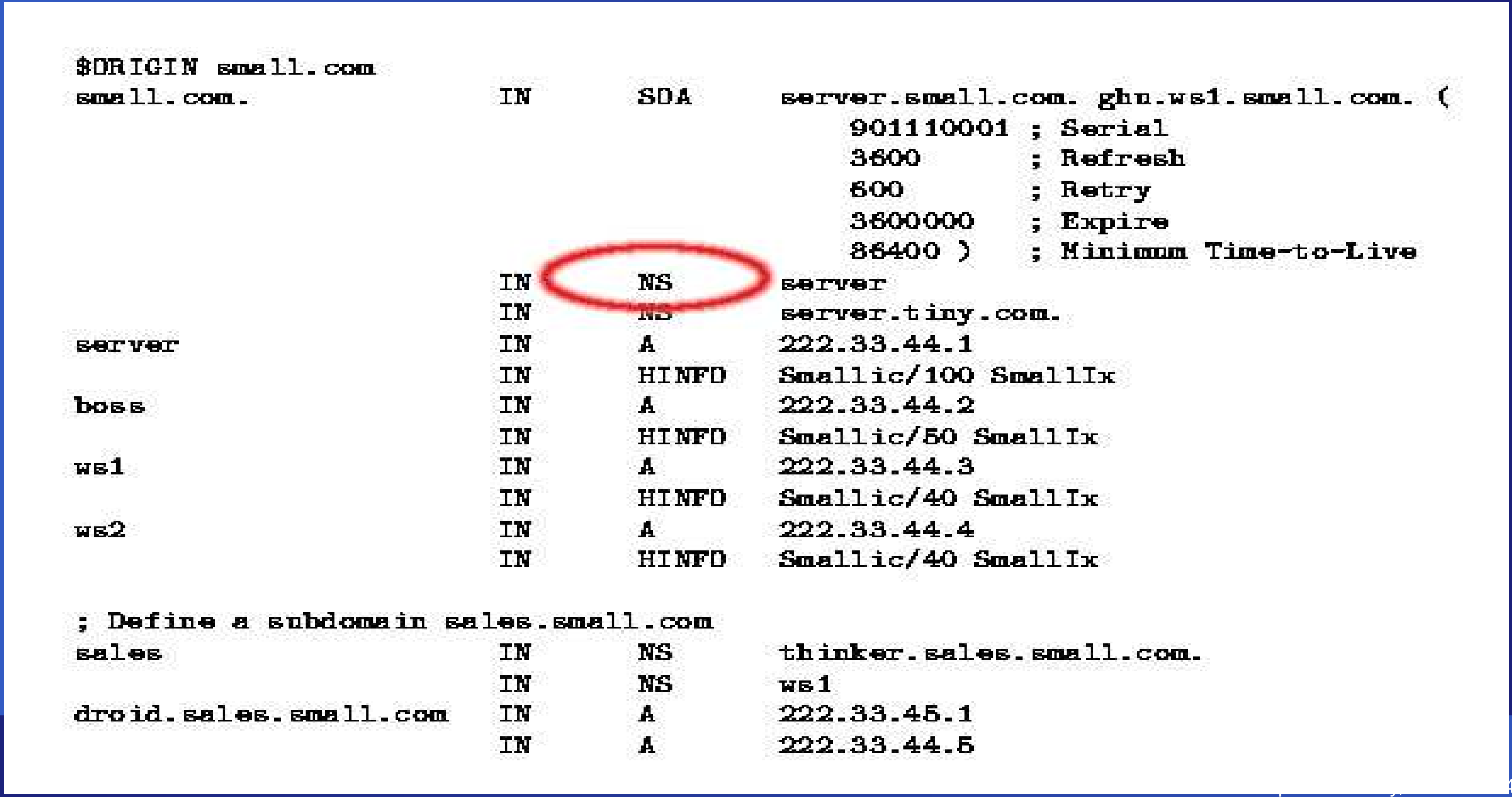


```
$ORIGIN small.com
small.com.              IN     SOA     server.small.com. ghu.ws1.small.com. (
                                          901110001 ; Serial
                                          3600      ; Refresh
                                          600       ; Retry
                                          3600000   ; Expire
                                          86400 )   ; Minimum Time-to-Live
                        IN     NS      server
                        IN     NS      server.tiny.com.
server                  IN     A       222.33.44.1
                        IN     HINFO   Smallic/100 SmallIx
boss                    IN     A       222.33.44.2
                        IN     HINFO   Smallic/50 SmallIx
ws1                     IN     A       222.33.44.3
                        IN     HINFO   Smallic/40 SmallIx
ws2                     IN     A       222.33.44.4
                        IN     HINFO   Smallic/40 SmallIx

; Define a subdomain sales.small.com
sales                   IN     NS      thinker.sales.small.com.
                        IN     NS      ws1
droid.sales.small.com   IN     A       222.33.45.1
                        IN     A       222.33.44.5
```

Zone Example 3/5

Name Server (NS):

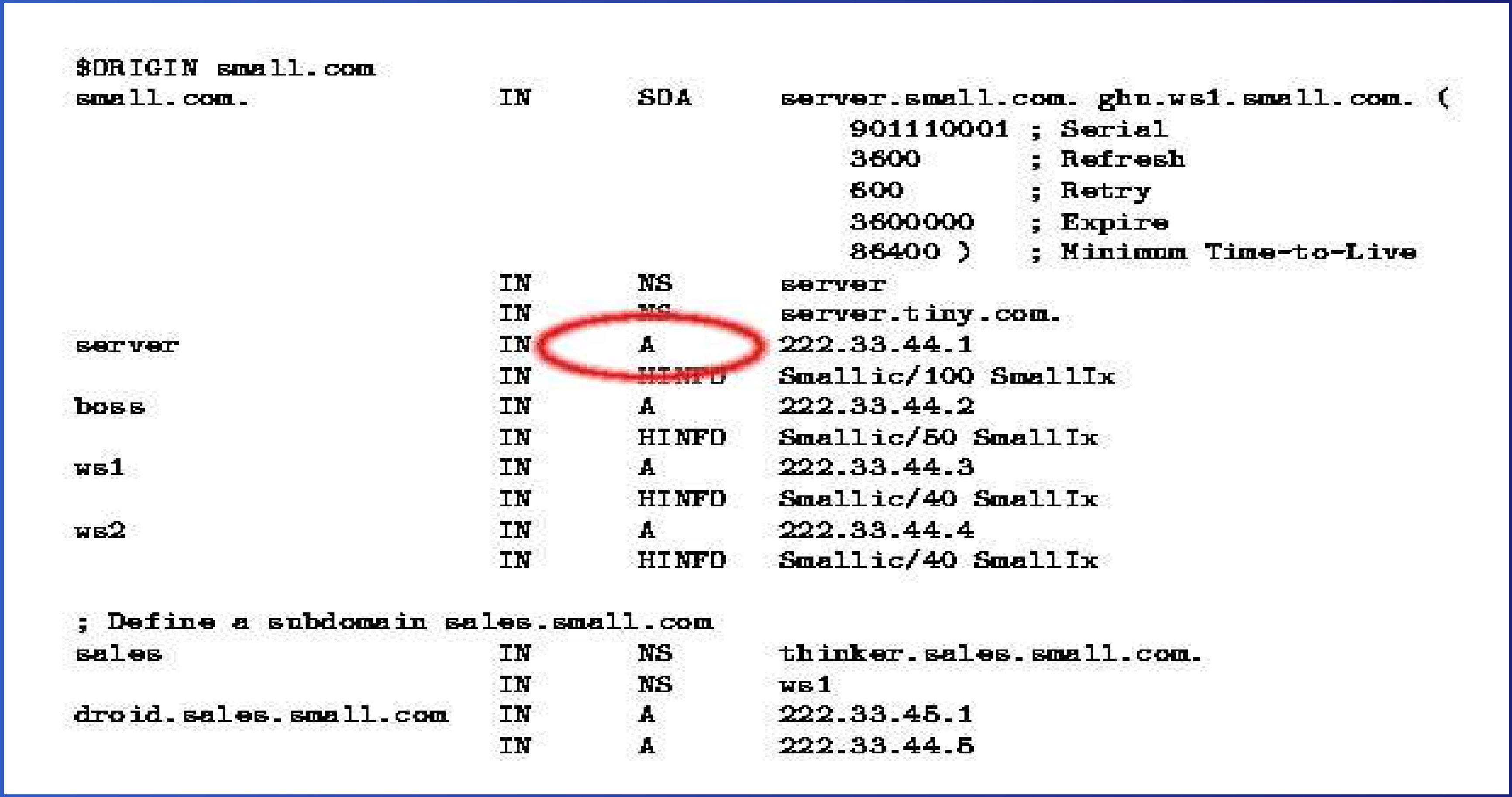
Specifies the authoritative name servers for the domain.



```
$ORIGIN small.com
small.com.              IN      SOA     server.small.com. ghu.ws1.small.com. (
                                            901110001 ; Serial
                                            3600      ; Refresh
                                            600       ; Retry
                                            3600000   ; Expire
                                            86400 )   ; Minimum Time-to-Live
                        IN      NS      server
                        IN      NS      server.tiny.com.
server                  IN      A       222.33.44.1
                        IN      HINFO   Smallic/100 SmallIx
boss                    IN      A       222.33.44.2
                        IN      HINFO   Smallic/50 SmallIx
ws1                     IN      A       222.33.44.3
                        IN      HINFO   Smallic/40 SmallIx
ws2                     IN      A       222.33.44.4
                        IN      HINFO   Smallic/40 SmallIx

; Define a subdomain sales.small.com
sales                   IN      NS      thinker.sales.small.com.
                        IN      NS      ws1
droid.sales.small.com   IN      A       222.33.45.1
                        IN      A       222.33.44.5
```

Zone Example 4/5

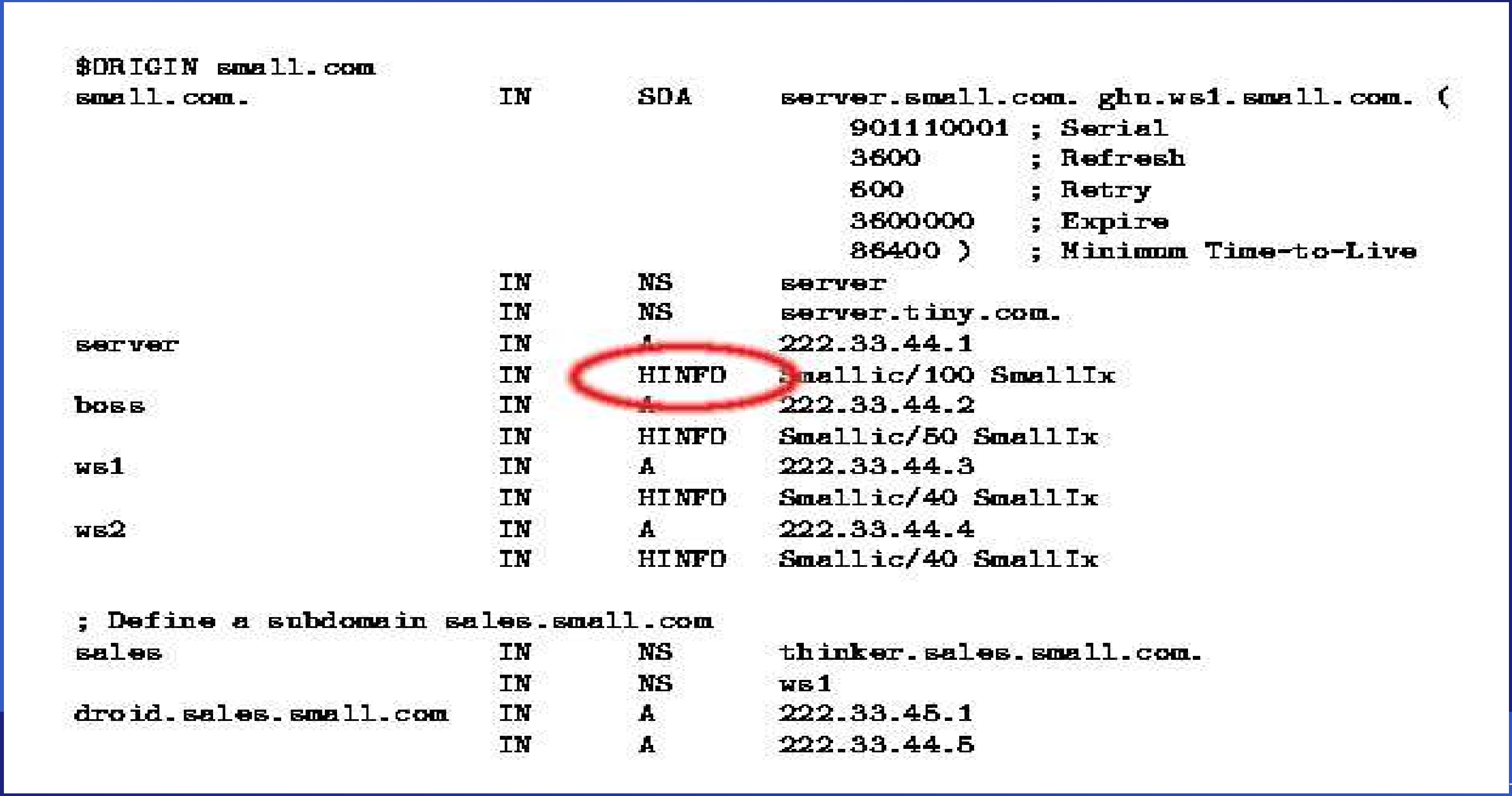
Address (A): Specifies the address of a host.



```
$ORIGIN small.com
small.com.              IN      SOA     server.small.com. ghu.ws1.small.com. (
                                            901110001 ; Serial
                                            3600      ; Refresh
                                            600       ; Retry
                                            3600000   ; Expire
                                            86400 )   ; Minimum Time-to-Live
                        IN      NS      server
                        IN      NS      server.tiny.com.
server                  IN      A       222.33.44.1
                        IN      HINFO   Smallic/100 SmallIx
boss                    IN      A       222.33.44.2
                        IN      HINFO   Smallic/50 SmallIx
ws1                     IN      A       222.33.44.3
                        IN      HINFO   Smallic/40 SmallIx
ws2                     IN      A       222.33.44.4
                        IN      HINFO   Smallic/40 SmallIx

; Define a subdomain sales.small.com
sales                   IN      NS      thinker.sales.small.com.
                        IN      NS      ws1
droid.sales.small.com   IN      A       222.33.45.1
                        IN      A       222.33.44.5
```

Zone Example 5/5

Host Info (HINFO): Specifies host information, like computer and operating system.



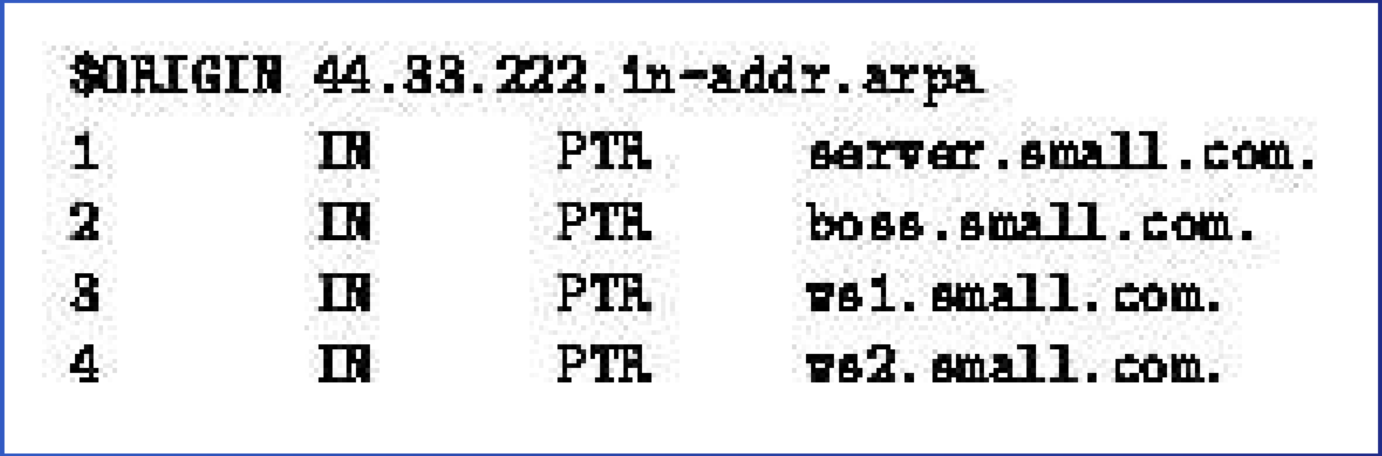
```
$ORIGIN small.com
small.com.              IN      SOA     server.small.com. ghu.ws1.small.com. (
                                            901110001 ; Serial
                                            3600      ; Refresh
                                            600       ; Retry
                                            3600000   ; Expire
                                            86400 )   ; Minimum Time-to-Live
                        IN      NS      server
                        IN      NS      server.tiny.com.
server                  IN      A       222.33.44.1
                        IN      HINFO   Smallic/100 SmallIx
boss                    IN      A       222.33.44.2
                        IN      HINFO   Smallic/50 SmallIx
ws1                     IN      A       222.33.44.3
                        IN      HINFO   Smallic/40 SmallIx
ws2                     IN      A       222.33.44.4
                        IN      HINFO   Smallic/40 SmallIx

; Define a subdomain sales.small.com
sales                   IN      NS      thinker.sales.small.com.
                        IN      NS      ws1
droid.sales.small.com   IN      A       222.33.45.1
                        IN      A       222.33.44.5
```

Forward queries

- Forward queries (asking for the IP address, providing a machine name) can be answered using the records from the zone.
- An item may also contain Additional Information, (e.g. providing NS *and* A records, when asked for the IP of an unknown host).

Inverse queries

- Inverse queries (asking for the machine name, providing an IP address) are answered using a separate, parallel tree, keyed by IP address.



```
$ORIGIN 44.33.222.in-addr.arpa
1         IN        PTR        server.small.com.
2         IN        PTR        boss.small.com.
3         IN        PTR        ws1.small.com.
4         IN        PTR        ws2.small.com.
```

Attack!

- Assumption: Attacker controlling a primary server for a DNS zone, including the inverse mapping tree, as well as all TCP port numbers.
- Attacker's goal: To find hosts that trust other hosts by name.
- Common examples:
  - Clusters of time-sharing machines.
  - File servers and their clients.

Starring:

- Softy, the victim:
  - `bullseye.softy.org 192.193.194.1`
  - `ringer.softy.org 192.193.194.64`
  - `groundzero.softy.org 192.193.194.65`
- Cuckoo, the attacker:
  - `cracker.ritts.org 150.151.152.153`

Guest star:

The vulnerability in the address-to-name mapping!

- Attacker changes the inverse mapping record for 150.151.152.153 from the correct cracker.ritts.org to ringer.softy.org
- Attacker attempts rlogin to bullseye.

- bullseye, the victim, validates the name of the calling machine:
    - It calls gethostbyaddr(), passing 150.151.152.153.
    - This generates a DNS inverse query for the PTR record for 153.152.151.150.in-addr.arpa
    - This retrieves ringer.softy.org
- Call accepted, attack succeeded.

Why?

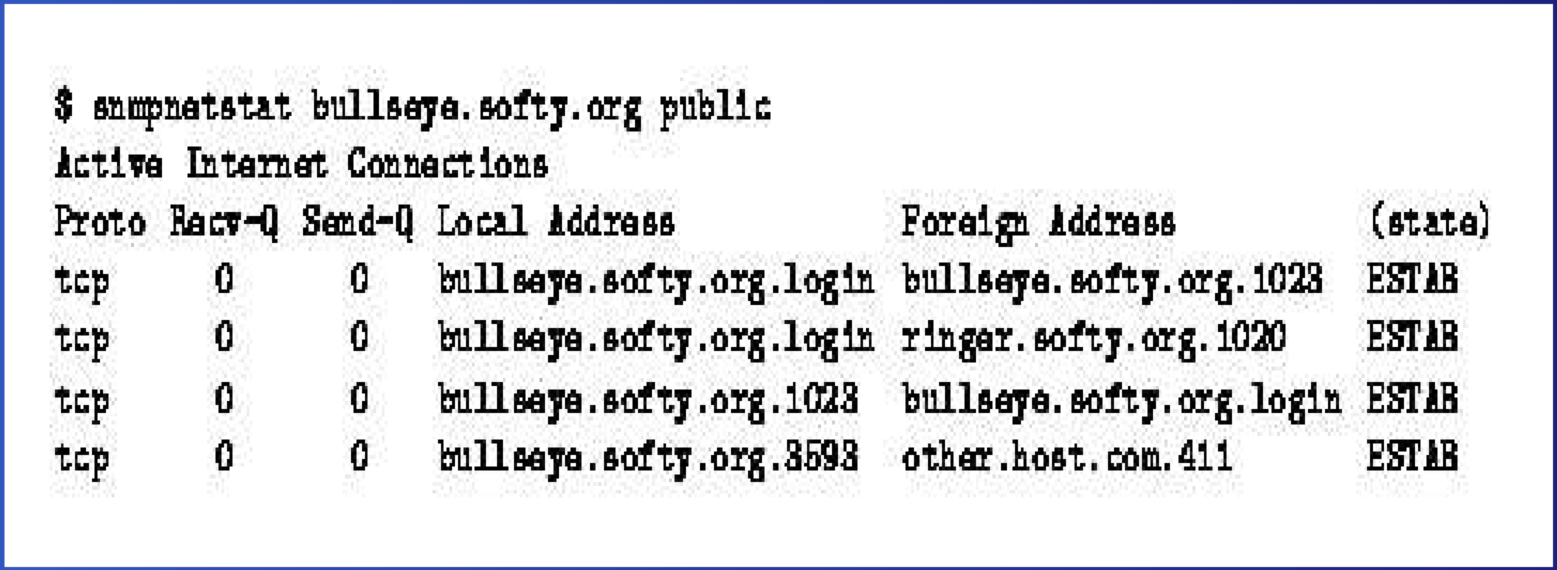
Because there is no forced linkage between the two DNS trees owned by Cuckoo, ritts.org and 152.151.150.in-addr.arpa, allowing the latter's entries to point to softy's hosts.

The rest are details...

- Finding a target host name.
- Finding a user name to impersonate.
- Finding a machine trusted by the target host.

SNMP abuse

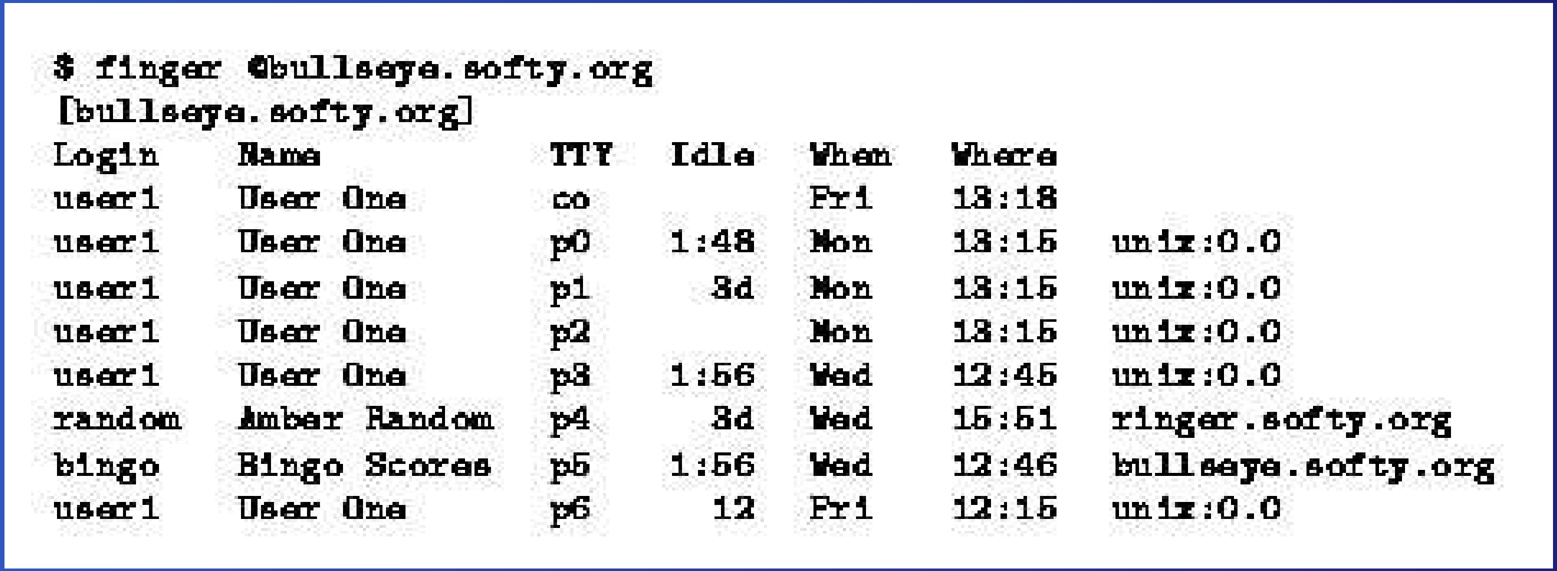
- Cuckoo finds the target host name from mail message or news article.
- He examines its TCP connection tables using SNMP.



```
$ snmpnetstat bullseye.softy.org public
Active Internet Connections
Proto Recv-Q Send-Q Local Address           Foreign Address         (state)
tcp        0      0 bullseye.softy.org.login bullseye.softy.org.1023 ESTAB
tcp        0      0 bullseye.softy.org.login ringer.softy.org.1020   ESTAB
tcp        0      0 bullseye.softy.org.1023  bullseye.softy.org.login ESTAB
tcp        0      0 bullseye.softy.org.3598  other.host.com.411      ESTAB
```

finger abuse

- He examines current users using finger.



```
$ finger @bullseye.softy.org
[bullseye.softy.org]
Login    Name           TTY   Idle   When   Where
user1    User One       co           Fri    13:18
user1    User One       p0    1:48   Mon    13:15   unix:0.0
user1    User One       p1      3d   Mon    13:15   unix:0.0
user1    User One       p2           Mon    13:15   unix:0.0
user1    User One       p3    1:56   Wed    12:45   unix:0.0
random   Amber Random   p4      3d   Wed    15:51   ringer.softy.org
bingo    Bingo Scores   p5    1:56   Wed    12:46   bullseye.softy.org
user1    User One       p6      12   Fri    12:15   unix:0.0
```

- He concludes: In bullseye, .rhosts file for bingo, authorizing user1 when coming from bullseye.

Done

- He modifies the appropriate PTR record.
- He creates local login names.
- He attacks.

Giving away information

Apart from SNMP and finger...

- e-mail,
- DNS (SOA records, zone transfers, HINFO records)
- SMTP
- FTP
- rpcinfo

...can also provide information about the victim.

The Berkeley fix

Validate the inverse mapping tree by looking at the corresponding node on the forward mapping tree.

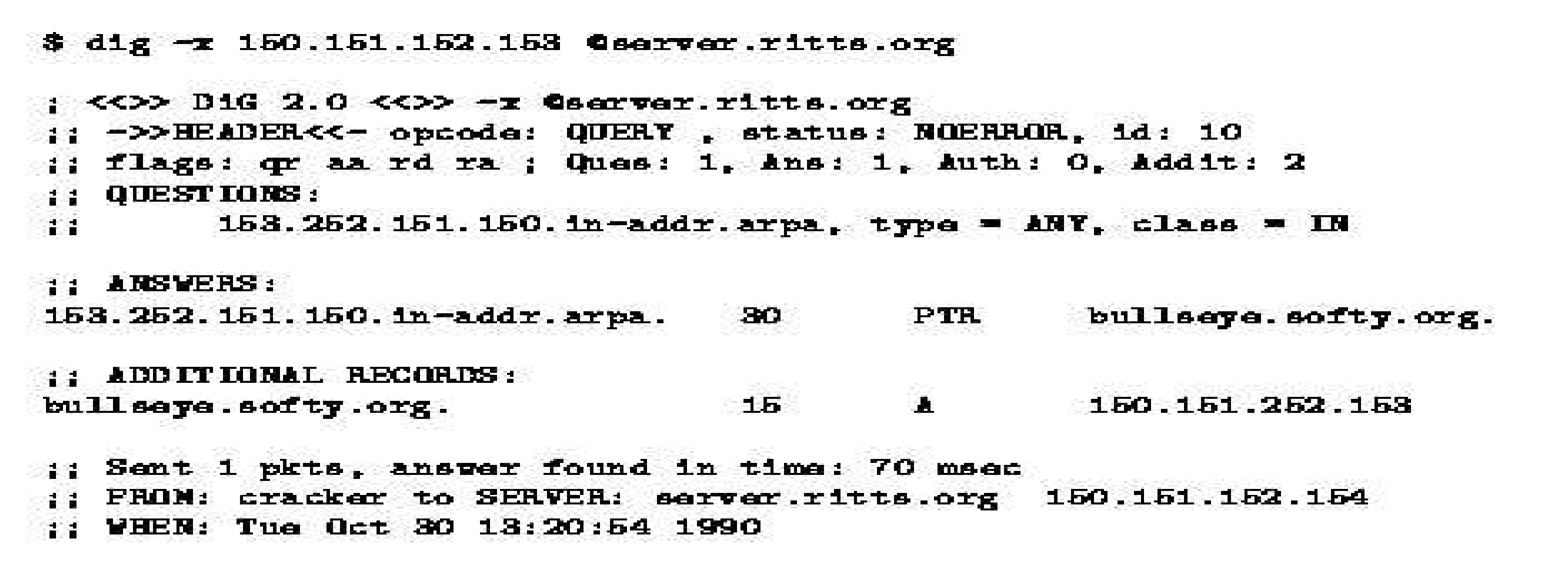
- If gethostbyaddr() returns bullseye.softy.org for 150.151.152.153, then gethostbyname() should return the same IP for the same name.
- Otherwise we have an impersonation.

How the fix is circumvented...

- The PTR record to answer gethostbyaddr()'s request is in Cuckoo's server.
- The A record to answer gethostbyname()'s request is in Softy's server.
- *However* the query might be answered by the local machine's name server cache.
- That DNS cache can be poisoned by the attacker...

Danger: Poison!

- The DNS message with the PTR record may contain a bogus A record in the Additional Information field (with short TTL).



```
$ dig -x 150.151.152.153 @server.ritts.org

; <<>> DiG 2.0 <<>> -x @server.ritts.org
;; ->>HEADER<<- opcode: QUERY , status: NOERROR, id: 10
;; flags: qr aa rd ra ; Ques: 1, Ans: 1, Auth: 0, Addit: 2
;; QUESTIONS:
;;        153.252.151.150.in-addr.arpa, type = ANY, class = IN

;; ANSWERS:
153.252.151.150.in-addr.arpa.    30       PTR      bullseye.softy.org.

;; ADDITIONAL RECORDS:
bullseye.softy.org.              15       A        150.151.252.153

;; Sent 1 pkts, answer found in time: 70 msec
;; FROM: cracker to SERVER: server.ritts.org  150.151.152.154
;; WHEN: Tue Oct 30 13:20:54 1990
```

- Or the bogus A record can be included in the NS records of a response to a lookup for a hostname

Therefore...

- Caching-only name servers are vulnerable!
- Authoritative name servers for a domain will reject updates for their zones.
- Hence they cannot be poisoned.
- But they are vulnerable for requests outside their zone.

Extra measures

- The target can act as a secondary server for the inverse mapping.
- The target can use a local mapping table like NIS before consulting DNS.

Hardening DNS Servers

- Bogus A records could be tracked back, if DNS server cache entries were tagged with their source.
- Additional Information could be used only in the specific context in which it was returned, and then discarded. (At a performance cost.)

Defenses

- Use cryptographic instead of name- or address-based authentication (e.g. Kerberos).
- Apart from Berkeley's fix:
  - Limit the trusted hosts to those for which the local machine has authoritative name information.
  - Have the local name server act as a secondary server for important neighboring zones, and thus possess authoritative forward-mapping data.
  - Have all machines possess definitive mapping information for the hosts within an organization.

Logging and Audtiing

- Attempts to impersonate hosts.
- Attempts to update authoritative zones.
- Attempts to connect to rlogind or rshd.
- Compare forward- and inverse-mapping data for a zone.

Abandon DNS?

- Return to static host tables?
  no (1990) NO! (2004)
- Problem lies not in DNS, but in inadequate host authentication methods.
- The information for host-to-address mapping is distributed, hence contamination from untrustworthy sources is always possible.
- The host table is huge and cannot be updated statically in a frequent and timely manner.

Is the attack still relevant?

- Paper written in 1990, published in 1995.
- 2004:
  - Name-based authentication is not that widely used anymore (ssh instead of rsh).
  - Firewalls disallow remote connections.
  - Too many BIND fixes since then.
  - Cryptographic authentication of DNS is used in experimental testbeds.
- Main idea still relevant, with new misuses.

DNS Threats in 2004

- Threat Analysis Of The Domain Name System. D. Atkins. IETF Draft (2003).
  - Packet Interception
  - ID Guessing and Query Prediction
  - Name Games
  - Betrayal By Trusted Server
  - Denial of Service
  - Authenticated Denial of Domain Names
  - Wildcards

DNSSEC

- DNS Security Extensions to provide end-to-end authenticity and integrity.
- All answers in DNSSEC are digitally signed.
- By checking the signature, a resolver is able to check if the info is identical (correct and complete) to the info on the authoritative server.
- D. Eastlake. RFC 2535 (1987).

Conclusions

- Inserting bogus resource records in a victim's DNS cache.
- Still possible.
- Luckily, name-based authentication is not that widely used anymore.
- However, other misuses like server redirection are equally grave.
- DNSSEC

References

1. P. Mockapetris. RFC 1034: Domain Concepts and Facilities. IETF, 1987.
2. P. Mockapetris. RFC 1035: Domain Implementation and Specification. IETF, 1987.
3. S. Bellovin. Using the Domain Name System for System Break-ins. USENIX, 1995.
4. P. Vixie. DNS and BIND security issues. USENIX, 1995.
5. C. Schuba and E. Spafford. Addressing weaknesses in the domain name system protocol. Master's thesis, 1993.
6. D. Eastlake. RFC 2535: Domain Name System Security Extensions. IETF, 1999.
7. D. Atkins. Internet Draft: Threat Analysis Of The Domain Name System. IETF, 2003.

Thank you!

Questions/comments?