DNS Security

Matt Beaumont-Gay
4/12/06

The Domain Name System

- Translates host names to network addresses
  - Also provides information on where to find services within a domain (MX, SRV records)
- Hierarchically organized
  - 13 root servers
  - Multiple servers for each top-level domain (e.g. com, edu, uk)
  - Each subsidiary domain associated with two or more authoritative servers

The Domain Name System

- Runs almost exclusively over UDP
  - Response associated to request via 16-bit ID
- End hosts configured with address for local resolver
- Local resolver configured with addresses of root nameservers
  - Almost always serves as a cache

DNS at work

- User points web browser at www.google.com
- Host's stub resolver sends recursive query to local nameserver
- Local nameserver asks...
  - a.root-servers.net about com
  - k.gtld-servers.net about google.com
  - ns2.google.com about www.google.com
- Local nameserver returns IP address to stub resolver

Why do we want to protect DNS?

- Q: What happens if the attacker can provide arbitrary response to end host's DNS queries?
- A: All sorts of evil
  - All responses point to adware/malware site
  - Response for bank address points to phishing site
  - All responses are NXDOMAIN or point to black hole address (denial of service)

Attack methods

- Packet interception
- Port or query ID guessing
  - 16 bits each
  - Each often predictable from previous queries or otherwise chosen from small subspace
- Cache poisoning
- Reconfigure end host so "local resolver" is a machine controlled by attacker

DNSSEC

- Cryptography provides fairly straightforward solution to problem of forged responses
- Nameservers sign responses
- Superzones provide public keys for subzones
  - Bootstrap off of well-known key for root servers
- Subtleties: negative responses, wildcard entries

Operational Issues

- The big issue: Nobody really uses DNSSEC
- Not deployed at root or (all but two) TLD nameservers
  - See http://www.dnssec-deployment.org/ for more info, including roadmap to high-level deployment
- Can be deployed by individual domains, but much less useful without chain of trust

Operational Issues

- Not deployed at clients
  - DNSSEC-aware stub resolver API not yet developed
  - Client needs either to have secure channel to local resolver or to verify response signatures itself to prevent spoofed responses
    - Deployment at local resolver prevents cache poisoning

Barriers to adoption

- Chicken-and-egg problem
  - Hopefully solved by push to deploy at root
- Zone enumeration via negative responses
  - Under discussion by IETF
- Computational cost of cryptography

DDoS via DNS

- DNS requests are unauthenticated
- Attacker sends requests w/ spoofed source address to some nameserver, nameserver sends replies to target
- With preparation, 60-byte request can elicit 4000-byte response (TXT record)

Questions

- How effective can partial deployment of DNSSEC be?
- Assuming DNSSEC becomes widely deployed, how will attackers adapt?
- What (besides forcing all sites to properly configure their resolvers) can be done about the DDoS amplification effect?