CHAPTER 1

FOOTPRINTING

Before the real fun for the hacker begins, three essential steps must be performed. This chapter will discuss the first one—*footprinting*—the fine art of gathering target information. For example, when thieves decide to rob a bank, they don't just walk in and start demanding money (not the smart ones, anyway). Instead, they take great pains in gathering information about the bank—the armored car routes and delivery times, the video cameras, and the number of tellers, escape exits, and anything else that will help in a successful misadventure.

The same requirement applies to successful attackers. They must harvest a wealth of information to execute a focused and surgical attack (one that won't be readily caught). As a result, attackers will gather as much information as possible about all aspects of an organization's security posture. Hackers end up with a unique *footprint* or profile of their Internet, remote access, and intranet/extranet presence. By following a structured methodology, attackers can systematically glean information from a multitude of sources to compile this critical footprint on any organization.

WHAT IS FOOTPRINTING?

The systematic footprinting of an organization enables attackers to create a complete profile of an organization's security posture. By using a combination of tools and techniques, attackers can take an unknown quantity (Widget Company's Internet connection) and reduce it to a specific range of domain names, network blocks, and individual IP addresses of systems directly connected to the Internet. While there are many types of footprinting techniques, they are primarily aimed at discovering information related to the following environments: Internet, intranet, remote access, and extranet. Table 1-1 depicts these environments and the critical information an attacker will try to identify.

Why Is Footprinting Necessary?

Footprinting is necessary to systematically and methodically ensure that all pieces of information related to the aforementioned technologies are identified. Without a sound methodology for performing this type of reconnaissance, you are likely to miss key pieces of information related to a specific technology or organization. Footprinting is often the most arduous task of trying to determine the security posture of an entity; however, it is one of the most important. Footprinting must be performed accurately and in a controlled fashion.

INTERNET FOOTPRINTING

While many footprinting techniques are similar across technologies (Internet and intranet), this chapter will focus on footprinting an organization's Internet connection(s). Remote access will be covered in detail in Chapter 9.

Technology	Identifies
Internet	Domain name Network blocks Specific IP addresses of systems reachable via the Internet TCP and UDP services running on each system identified System architecture (for example, SPARC vs. X86) Access control mechanisms and related access control lists (ACLs) Intrusion detection systems (IDSes) System enumeration (user and group names, system banners, routing tables, SNMP information)
Intranet	Networking protocols in use (for example, IP, IPX, DecNET, and so on) Internal domain names Network blocks Specific IP addresses of systems reachable via intranet TCP and UDP services running on each system identified System architecture (for example, SPARC vs. X86) Access control mechanisms and related access control lists (ACLs) Intrusion detection systems System enumeration (user and group names, system banners, routing tables, SNMP information)
Remote access	Analog/digital telephone numbers Remote system type Authentication mechanisms VPNs and related protocols (IPSEC, PPTP)
Extranet	Connection origination and destination Type of connection Access control mechanism

Table 1-1. Environments and the Critical Information Attackers Can Identify

It is difficult to provide a step-by-step guide on footprinting because it is an activity that may lead you down several paths. However, this chapter delineates basic steps that should allow you to complete a thorough footprint analysis. Many of these techniques can be applied to the other technologies mentioned earlier.

Step 1. Determine the Scope of Your Activities

The first item to address is to determine the scope of your footprinting activities. Are you going to footprint an entire organization, or are you going to limit your activities to certain locations (for example, corporate vs. subsidiaries)? In some cases, it may be a daunting task to determine all the entities associated with a target organization. Luckily, the Internet provides a vast pool of resources you can use to help narrow the scope of activities and also provides some insight as to the types and amount of information publicly available about your organization and its employees.

Open Source Search

Popularity:	*9*
Simplicity:	*9*
Impact:	*2*
Risk Rating:	***7***

As a starting point, peruse the target organization's web page if they have one. Many times an organization's web page provides a ridiculous amount of information that can aid attackers. We have actually seen organizations list security configuration options for their firewall system directly on their Internet web server. Other items of interest include

- Locations
- Related companies or entities
- Merger or acquisition news
- Phone numbers
- Contact names and email addresses
- Privacy or security policies indicating the types of security mechanisms in place
- Links to other web servers related to the organization

In addition, try reviewing the HTML source code for comments. Many items not listed for public consumption are buried in HTML comment tags such as "<," "!," and "--." Viewing the source code offline may be faster than viewing it online, so it is often beneficial to mirror the entire site for offline viewing. Having a copy of the site locally may allow you to programmatically search for comments or other items of interest, thus making your footprinting activities more efficient. `Wget` (http://www.gnu.org/software/

wget/wget.html) for UNIX and Teleport Pro (http://www.tenmax.com/teleport/home.htm) for Windows are great utilities to mirror entire web sites.

After studying web pages, you can perform open source searches for information relating to the target organization. News articles, press releases, and so on, may provide additional clues about the state of the organization and their security posture. Web sites such as finance.yahoo.com or http://www.companysleuth.com provide a plethora of information. If you are profiling a company that is mostly Internet based, you may find by searching for related news stories that they have had numerous security incidents. Using your web search engine of choice will suffice for this activity. However, there are more advanced searching tools and criteria you can use to uncover additional information.

The FerretPRO suite of search tools from FerretSoft (http://www.ferretsoft.com) is one of our favorites. WebFerretPRO enables you to search many different search engines simultaneously. In addition, other tools in the suite allow you to search IRC, USENET, email, and file databases looking for clues. Also, if you're looking for a free solution to search multiple search engines, check out http://www.dogpile.com.

Searching USENET for postings related to *@example*.com often reveals useful information. In one case, we saw a posting from a system administrator's work account regarding his new PBX system. He said this switch was new to him, and he didn't know how to turn off the default accounts and passwords. We'd hate to guess how many phone phreaks were salivating over the prospect of making free calls at that organization. Needless to say, you can gain additional insight into the organization and the technical prowess of its staff just by reviewing their postings.

Lastly, you can use the advanced searching capabilities of some of the major search engines like AltaVista or Hotbot. These search engines provide a handy facility that allows you to search for all sites that have links back to the target organization's domain. This may not seem significant at first, but let's explore the implications. Suppose someone in an organization decides to put up a rogue web site at home or on the target network's site. This web server may not be secure or sanctioned by the organization. So we can begin to look for potential rogue web sites just by determining which sites actually link to the target organization's web server, as shown in Figure 1-1.

You can see that the search returned all sites that link back to http://www.l0pht.com and that contain the word "hacking." So you could easily use this search facility to find sites linked to your target domain.

The last example, depicted in Figure 1-2, allows you to limit your search to a particular site. In our example, we searched http://www.l0pht.com for all occurrences of "mudge." This query could easily be modified to search for other items of interest.

Obviously, these examples don't cover every conceivable item to search for during your travels—be creative. Sometimes the most outlandish search yields the most productive results.

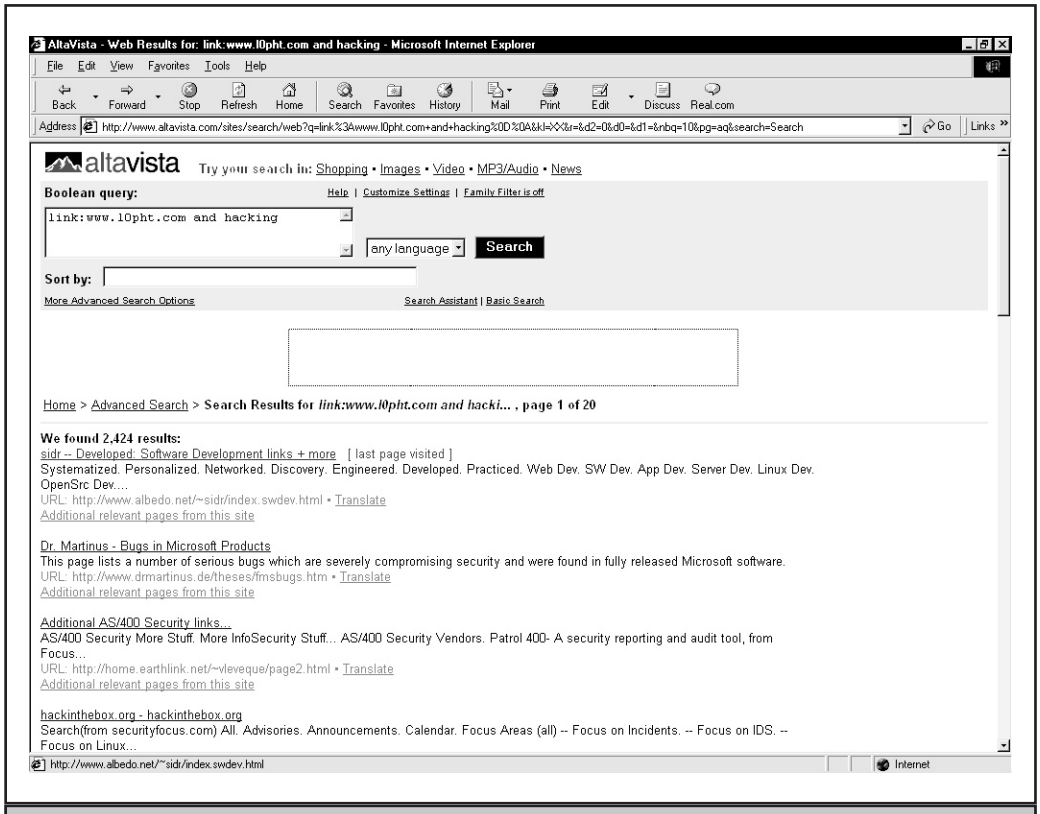

Figure 1-1. With the AltaVista search engine, use the `link:www.example.com` directive to query all sites with links back to the target domain.

EDGAR Search

For targets that are publicly traded companies, you can consult the Securities and Exchange Commission (SEC) EDGAR database at http://www.sec.gov, as shown in Figure 1-3.

One of the biggest problems organizations have is managing their Internet connections, especially when they are actively acquiring or merging with other entities. So it is important to focus on newly acquired entities. Two of the best SEC publications to review are the 10-Q and 10-K. The 10-Q is a quick snapshot of what the organization has done over the last quarter. This update includes the purchase or disposition of other entities. The 10-K is a yearly update of what the company has done and may not be as timely as the 10-Q. It is a good idea to peruse these documents by searching for "subsidiary" or "subsequent events." This may provide you with information on a newly acquired entity. Often organizations will scramble to connect the acquired entities to their corporate network with little regard for security. So it is likely that you may be able to find security weaknesses

Figure 1-2. With AltaVista, use the `host:`*`example`*`.com` directive to query the site for the specified string (for example, "mudge").

in the acquired entity that would allow you to leapfrog into the parent company. Attackers are opportunistic and are likely to take advantage of the chaos that normally comes with combining networks.

With an EDGAR search, keep in mind that you are looking for entity names that are different from the parent company. This will become critical in subsequent steps when you perform organizational queries from the various whois databases available (see "Step 2. Network Enumeration").

Countermeasure: Public Database Security

Much of the information discussed earlier must be made publicly available; this is especially true for publicly traded companies. However, it is important to evaluate and classify the type of information that is publicly disseminated. The Site Security Handbook (RFC 2196) can be found at http://www.ietf.org/rfc/rfc2196.txt and is a wonderful resource

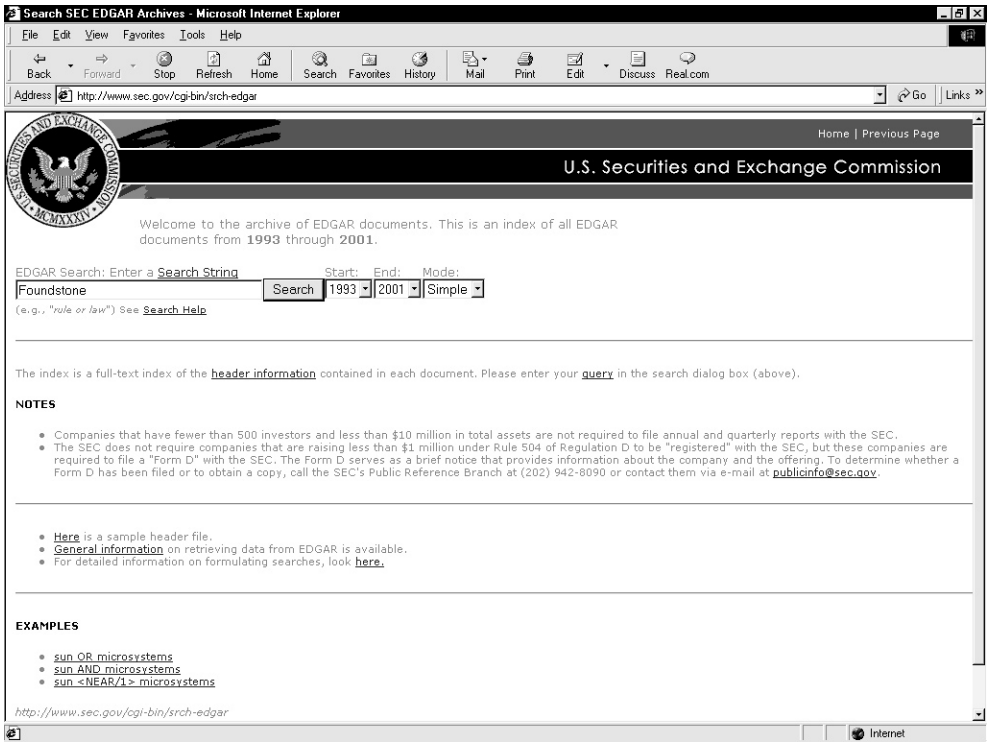

Figure 1-3. The EDGAR database allows you to query public documents, providing important insight into the breadth of the organization by identifying its associated entities.

for many policy-related issues. Finally, remove any unnecessary information from your web pages that may aid an attacker in gaining access to your network.

Step 2. Network Enumeration

Popularity:	*9*
Simplicity:	*9*
Impact:	*5*
Risk Rating:	***8***

The first step in the network enumeration process is to identify domain names and associated networks related to a particular organization. Domain names represent the

company's presence on the Internet and are the Internet equivalent to your company's name, such as "AAAApainting.com" and "moetavern.com."

To enumerate these domains and begin to discover the networks attached to them, you must scour the Internet. There are multiple whois databases you can query that will provide a wealth of information about each entity we are trying to footprint. Before the end of 1999, Network Solutions had a monopoly as the main registrar for domain names (com, net, edu, and org) and maintained this information on their whois servers. This monopoly was dissolved and currently there is a multitude of accredited registrars (http://www.internic.net/alpha.html). Having new registrars available adds steps in finding our targets (see "Registrar Query" later in this step). We will need to query the correct registrar for the information we are looking for.

There are many different mechanisms (see Table 1-2) to query the various whois databases. Regardless of the mechanism, you should still receive the same information. Users should consult Table 1-3 for other whois servers when looking for domains other than com, net, edu, or org. Another valuable resource, especially for finding whois servers outside of the United States, is http://www.allwhois.com. This is one of the most complete whois resources on the Internet.

Mechanism	Resources	Platform
Web interface	http://www.networksolutions.com/ http://www.arin.net	Any platform with a web client
Whois client	Whois is supplied with most versions of UNIX. Fwhois was created by Chris Cappuccio <ccappuc@santefe.edu>	UNIX
WS_Ping ProPack	http://www.ipswitch.com/	Windows 95/NT/2000
Sam Spade	http://www.samspade.org/ssw	Windows 95/NT/2000
Sam Spade Web Interface	http://www.samspade.org/	Any platform with a web client
Netscan tools	http://www.netscantools.com/nstpromain.html	Windows 95/NT/2000
Xwhois	http://c64.org/~nr/xwhois/	UNIX with X and GTK+ GUI toolkit

Table 1-2. Whois Searching Techniques and Data Sources

Whois Server	Addresses
European IP Address Allocations	http://www.ripe.net/
Asia Pacific IP Address Allocations	http://whois.apnic.net
U.S. military	http://whois.nic.mil
U.S. government	http://whois.nic.gov

Table 1-3. Government, Military, and International Sources of Whois Databases

Different information can be gleaned with each query. The following query types provide the majority of information hackers use to begin their attack:

- **Registrar** Displays specific registrar information and associated whois servers
- **Organizational** Displays all information related to a particular organization
- **Domain** Displays all information related to a particular domain
- **Network** Displays all information related to a particular network or a single IP address
- **Point of contact (POC)** Displays all information related to a specific person, typically the administrative contact

Registrar Query

With the advent of the shared registry system (that is, multiple registrars), we must consult the whois.crsnic.net server to obtain a listing of potential domains that match our target and their associated registrar information. We need to determine the correct registrar so that we can submit detailed queries to the correct database in subsequent steps. For our example, we will use "Acme Networks" as our target organization and perform our query from a UNIX (Red Hat 6.2) command shell. In the version of `whois` we are using, the @ option allows you to specify an alternate database. In some BSD-derived `whois` clients (for example, OpenBSD or FreeBSD), it is possible to use the `-a` option to specify an alternate database. You should `man whois` for more information on how to submit whois queries with your `whois` client.

It is advantageous to use a wildcard when performing this search because it will provide additional search results. Using a "." after "acme" will list all occurrences of domains that begin with "acme" rather than domains that simply match "acme" exactly. In addition, consult http://www.networksolutions.com/en_US/help/whoishelp.html for additional information on submitting advanced searches. Many of the hints contained in this document can help you dial-in your search with much more precision.

```
[bash]$ whois "acme."@whois.crsnic.net
[whois.crsnic.net]
Whois Server Version 1.1

Domain names in the .com, .net, and .org domains can now be registered
with many different competing registrars. Go to http://www.internic.net
for detailed information.

ACMETRAVEL.COM
ACMETECH.COM
ACMES.COM
ACMERACE.NET
ACMEINC.COM
ACMECOSMETICS.COM
ACME.ORG
ACME.NET
ACME.COM
ACME-INC.COM
```

If we are interested in obtaining more information on acme.net, we can continue to drill down further to determine the correct registrar.

```
[[bash]$ whois "acme.net"@whois.crsnic.net
Whois Server Version 1.1

Domain names in the .com, .net, and .org domains can now be registered
with many different competing registrars. Go to http://www.internic.net
for detailed information.

   Domain Name: ACME.NET
   Registrar: NETWORK SOLUTIONS, INC.
   Whois Server: whois.networksolutions.com
   Referral URL: www.networksolutions.com
   Name Server: DNS1.ACME.NET
   Name Server: DNS2.ACME.NET
```

We can see that Network Solutions is the registrar for this organization, which is quite common for any organization on the Internet before adoption of the shared registry system. For subsequent queries, we must query the respective registrar's database because they maintain the detailed information we want.

Organizational Query

Once we have identified a registrar, we can submit an organizational query. This type of query will search a specific registrar for all instances of the entity name and is broader

than looking for just a domain name. We must use the keyword "name" and submit the query to Network Solutions.

```
[bash]$ whois "name Acme Networks"@whois.networksolutions.com
Acme Networks (NAUTILUS-AZ-DOM) NAUTILUS-NJ.COM
Acme Networks (WINDOWS4-DOM)                        WINDOWS.NET
Acme Networks (BURNER-DOM)                          BURNER.COM
Acme Networks (ACME2-DOM)                           ACME.NET
Acme Networks (RIGHTBABE-DOM)                       RIGHTBABE.COM
Acme Networks (ARTS2-DOM)                           ARTS.ORG
Acme Networks (HR-DEVELOPMENT-DOM)                  HR-DEVELOPMENT.COM
Acme Networks (NTSOURCE-DOM)                        NTSOURCE.COM
Acme Networks (LOCALNUMBER-DOM)                     LOCALNUMBER.NET
Acme Networks (LOCALNUMBERS2-DOM)                   LOCALNUMBERS.NET
Acme Networks (Y2MAN-DOM)                           Y2MAN.COM
Acme Networks (Y2MAN2-DOM)                          Y2MAN.NET
Acme Networks for Christ Hospital (CHOSPITAL-DOM) CHOSPITAL.ORG
...
```

From this, we can see many different domains are associated with Acme Networks. However, are they real networks associated with those domains, or have they been registered for future use or to protect a trademark? We need to continue drilling down until we find a live network.

When you are performing an organizational query for a large organization, there may be hundreds or thousands of records associated with it. Before spamming became so popular, it was possible to download the entire com domain from Network Solutions. Knowing this, Network Solutions whois servers will truncate the results and only display the first 50 records.

Domain Query

Based on our organizational query, the most likely candidate to start with is the Acme.net domain since the entity is Acme Networks. (Of course, all real names and references have been changed.)

```
[bash]$ whois acme.net@whois.networksolutions.com

[whois.networksolutions.com]
Registrant:

Acme Networks (ACME2-DOM)
11 Town Center Ave.
Einstein, AZ 21098

Domain Name: ACME.NET
```

```
Administrative Contact, Technical Contact, Zone Contact:
Boyd, Woody [Network Engineer]  (WB9201)  woody@ACME.NET
201-555-9011 (201)555-3338 (FAX) 201-555-1212

Record last updated on 13-Sep-95.
Record created on 30-May-95.
Database last updated on 14-Apr-99 13:20:47 EDT.

Domain servers in listed order:
   DNS.ACME.NET           10.10.10.1
   DNS2.ACME.NET          10.10.10.2
```

This type of query provides you with information related to the following:

- The registrant
- The domain name
- The administrative contact
- When the record was created and updated
- The primary and secondary DNS servers

At this point, you need to become a bit of a cybersleuth. Analyze the information for clues that will provide you with more information. We commonly refer to excess information or information leakage as "enticements." That is, they may entice an attacker into mounting a more focused attack. Let us review this information in detail.

By inspecting the registrant information, we can ascertain if this domain belongs to the entity that we are trying to footprint. We know that Acme Networks is located in Arizona, so it is safe to assume this information is relevant to our footprint analysis. Keep in mind, the registrant's locale doesn't necessarily have to correlate to the physical locale of the entity. Many entities have multiple geographic locations, each with its own Internet connections; however, they may all be registered under one common entity. For your domain, it would be necessary to review the location and determine if it was related to your organization. The domain name is the same domain name that we used for our query, so this is nothing new to us.

The administrative contact is an important piece of information because it may tell you the name of the person responsible for the Internet connection or firewall. It also lists voice and fax numbers. This information is an enormous help when you're performing a dial-in penetration review. Just fire up the wardialers in the noted range, and you're off to a good start in identifying potential modem numbers. In addition, an intruder will often pose as the administrative contact, using social engineering on unsuspecting users in an organization. An attacker will send spoofed email messages posing as the administrative contact to a gullible user. It is amazing how many users will change their password to whatever you like, as long as it looks like the request is being sent from a trusted technical support person.

The record creation and modification dates indicate how accurate the information is. If the record was created five years ago but hasn't been updated since, it is a good bet some of the information (for example, Administrative Contact) may be out of date.

The last piece of information provides you with the authoritative DNS servers. The first one listed is the primary DNS server, and subsequent DNS servers will be secondary, tertiary, and so on. We will need this information for our DNS interrogation discussed later in this chapter. Additionally, we can try to use the network range listed as a starting point for our network query of the ARIN database.

TIP Using a `server` directive with the HST record gained from a whois query, you can discover the other domains for which a given DNS server is authoritative. The following steps show you how.

1. Execute a domain query as detailed earlier.
2. Locate the first DNS server.
3. Execute a whois query on that DNS server:

   ```
   whois "HOST 10.10.10.1"@whois.networksolutions.com
   ```

4. Locate the HST record for the DNS server.
5. Execute a whois query with the server directive using `whois` and the respective HST record:

   ```
   whois "SERVER NS9999-HST"@whois.networksolutions.com
   ```

Network Query

The American Registry for Internet Numbers (ARIN) is another database that we can use to determine networks associated with our target domain. This database maintains specific network blocks that an organization owns. It is particularly important to perform this search to determine if a system is actually owned by the target organization or if it is being co-located or hosted by another organization such as an ISP.

In our example, we can try to determine all the networks that "Acme Networks" owns. Querying the ARIN database is a particularly handy query because it is not subject to the 50-record limit implemented by Network Solutions. Note the use of the "." wildcard.

```
[bash]$ whois "Acme Net."@whois.arin.net
[whois.arin.net]
Acme Networks (ASN-XXXX)     XXXX                  99999
Acme Networks (NETBLK)       10.10.10.0 – 10.20.129.255
```

A more specific query can be submitted based upon a particular net block (10.10.10.0).

```
[bash]$ whois 10.10.10.0@whois.arin.net
[whois.arin.net]
```

```
Major ISP USA (NETBLK-MI-05BLK) MI-05BLK    10.10.0.0 - 10.30.255.255
ACME NETWORKS, INC. (NETBLK-MI-10-10-10) CW-10-10-10
    10.10.10.0 - 10.20.129.255
```

ARIN provides a handy web-based query mechanism, as shown in Figure 1-4. By reviewing the output, we can see that "Major ISP USA" is the main backbone provider and has assigned a class A network (see *TCP/IP Illustrated Volume 1* by Richard Stevens for a complete discussion of TCP/IP) to Acme Networks. Thus, we can conclude that this is a valid network owned by Acme Networks.

POC Query

Since the administrative contact may be the administrative contact for multiple organizations, it is advantageous to perform a point of contact (POC) query to search by the user's

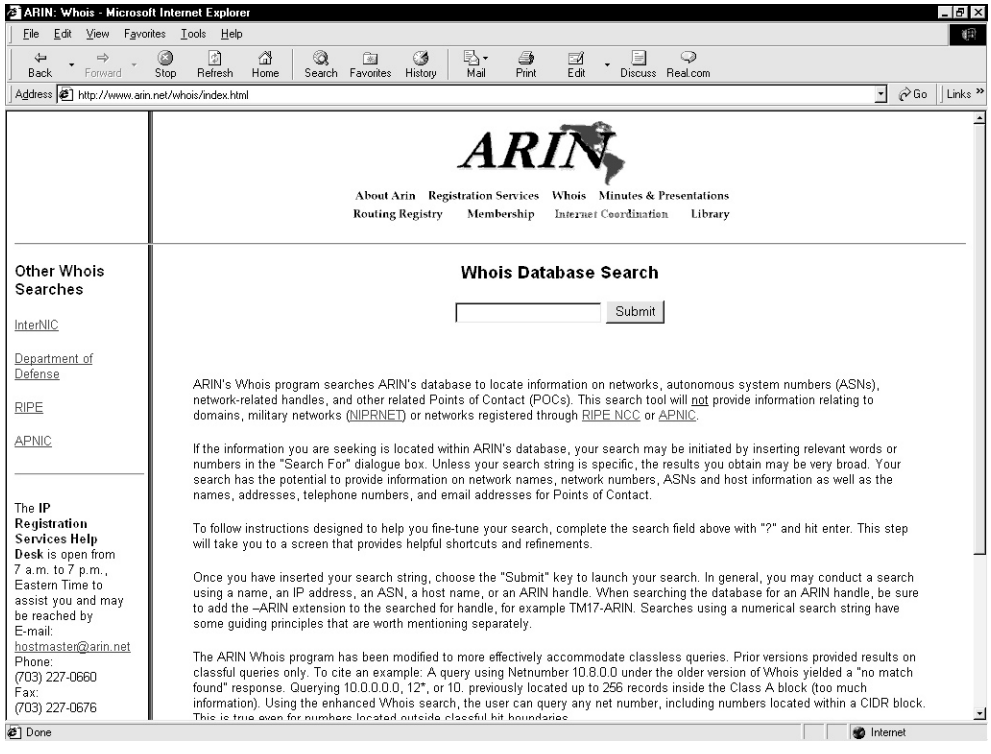

Figure 1-4. One of the easiest ways to search for ARIN information is from their web site.

database handle. The handle we are searching for is "WB9201," derived from the preceding domain query. You may uncover a domain that you were unaware of.

```
[bash]$ whois "HANDLE WB9201"@whois.networksolutions.com
Boyd, Woody [Network Engineer] (WB9201)             woody@ACME.NET
   BIG ENTERPRISES
   11 TOWN CENTER AVE
   EINSTEIN, AZ 20198
   201-555-1212 (201)555-1212 (FAX) 201-555-1212
```

We could also search for @Acme.net to obtain a listing of all mail addresses for a given domain. We have truncated the following results for brevity:

```
[bash]$ whois "@acme.net"@whois.networksolutions.net
Smith, Janet (JS9999)    jsmith@ACME.NET   (201)555-9211 (FAX) (201)555-3643
Benson, Bob  (BB9999)       bob@ACME.NET   (201)555-0988
Manual, Eric(EM9999)      ericm@ACME.NET   (201)555-8484 (FAX) (201)555-8485
Bixon, Rob (RB9999)      rbixon@ACME.NET   (201)555-8072
```

Countermeasure: Public Database Security

Much of the information contained in the various databases discussed thus far is geared at public disclosure. Administrative contacts, registered net blocks, and authoritative name server information is required when an organization registers a domain on the Internet. However, security considerations should be employed to make the job of attackers much more difficult.

Many times an administrative contact will leave an organization and still be able to change the organization's domain information. Thus, first ensure that the information listed in the database is accurate. Update the administrative, technical, and billing contact information as necessary. Furthermore, consider the phone numbers and addresses listed. These can be used as a starting point for a dial-in attack or for social engineering purposes. Consider using a toll-free number or a number that is not in your organization's phone exchange. In addition, we have seen several organizations list a fictitious administrative contact, hoping to trip up a would-be social engineer. If any employee receives an email or calls to or from the fictitious contact, it may tip off the information security department that there is a potential problem.

Another hazard with domain registration arises from the way that some registrars allow updates. For example, the current Network Solutions implementation allows automated online changes to domain information. Network Solutions authenticates the domain registrant's identity through three different methods: the FROM field in an email, a password, or via a Pretty Good Privacy (PGP) key. Shockingly, the default authentication method is the FROM field via email. The security implications of this authentication mechanism are prodigious. Essentially, anyone can trivially forge an email address and change the information associated with your domain, better known as *domain hijacking*. This is exactly what happened to AOL on October 16, 1998, as reported by the *Washington Post*. Someone impersonated an AOL official and changed AOL's domain information so that all traffic was

directed to autonete.net. AOL recovered quickly from this incident, but it underscores the fragility of an organization's presence on the Internet. It is important to choose a more secure solution like password or PGP authentication to change domain information. Moreover, the administrative or technical contact is required to establish the authentication mechanism via Contact Form from Network Solutions.

Step 3. DNS Interrogation

After identifying all the associated domains, you can begin to query the DNS. DNS is a distributed database used to map IP addresses to hostnames and vice versa. If DNS is configured insecurely, it is possible to obtain revealing information about the organization.

Zone Transfers

Popularity:	*9*
Simplicity:	*9*
Impact:	*3*
Risk Rating:	***7***

One of the most serious misconfigurations a system administrator can make is allowing untrusted Internet users to perform a DNS zone transfer.

A *zone transfer* allows a secondary master server to update its zone database from the primary master. This provides for redundancy when running DNS, should the primary name server become unavailable. Generally, a DNS zone transfer only needs to be performed by secondary master DNS servers. Many DNS servers, however, are misconfigured and provide a copy of the zone to anyone who asks. This isn't necessarily bad if the only information provided is related to systems that are connected to the Internet and have valid hostnames, although it makes it that much easier for attackers to find potential targets. The real problem occurs when an organization does not use a public/private DNS mechanism to segregate their external DNS information (which is public) from its internal, private DNS information. In this case, internal hostnames and IP addresses are disclosed to the attacker. Providing internal IP address information to an untrusted user over the Internet is akin to providing a complete blueprint, or roadmap, of an organization's internal network.

Let's take a look at several methods we can use to perform zone transfers and the types of information that can be gleaned. While there are many different tools to perform zone transfers, we are going to limit the discussion to several common types.

A simple way to perform a zone transfer is to use the `nslookup` client that is usually provided with most UNIX and NT implementations. We can use `nslookup` in interactive mode as follows:

```
[bash]$ nslookup
Default Server:  dns2.acme.net
Address:  10.10.20.2
```

```
>> server 10.10.10.2

Default Server:  [10.10.10.2]
Address: 10.10.10.2

>> set type=any
>> ls -d Acme.net. >> /tmp/zone_out
```

We first run `nslookup` in interactive mode. Once started, it will tell you the default name server that it is using, which is normally your organization's DNS server or a DNS server provided by your Internet service provider (ISP). However, our DNS server (10.10.20.2) is not authoritative for our target domain, so it will not have all the DNS records we are looking for. Thus, we need to manually tell `nslookup` which DNS server to query. In our example, we want to use the primary DNS server for Acme Networks (10.10.10.2). Recall that we found this information from our domain whois lookup performed earlier.

Next we set the record type to *any*. This will allow you to pull any DNS records available (`man nslookup`) for a complete list.

Finally, we use the `ls` option to list all the associated records for the domain. The `-d` switch is used to list all records for the domain. We append a "." to the end to signify the fully qualified domain name—however, you can leave this off most times. In addition, we redirect our output to the file `/tmp/zone_out` so that we can manipulate the output later.

After completing the zone transfer, we can view the file to see if there is any interesting information that will allow us to target specific systems. Let's review the output:

```
[bash]$ more zone_out
acct18                  1D IN A         192.168.230.3
                        1D IN HINFO     "Gateway2000" "WinWKGRPS"
                        1D IN MX        0 acmeadmin-smtp
                        1D IN RP        bsmith.rci bsmith.who
                        1D IN TXT       "Location:Telephone Room"
ce                      1D IN CNAME     aesop
au                      1D IN A         192.168.230.4
                        1D IN HINFO     "Aspect" "MS-DOS"
                        1D IN MX        0 andromeda
                        1D IN RP        jcoy.erebus jcoy.who
                        1D IN TXT       "Location: Library"
acct21                  1D IN A         192.168.230.5
                        1D IN HINFO     "Gateway2000" "WinWKGRPS"
                        1D IN MX        0 acmeadmin-smtp
                        1D IN RP        bsmith.rci bsmith.who
                        1D IN TXT       "Location:Accounting"
```

We won't go through each record in detail, but we will point out several important types. We see that for each entry we have an *A* record that denotes the IP address of the system name located to the right. In addition, each host has an HINFO record that identifies the platform or type of operating system running (see RFC 952). HINFO records are

not needed, but provide a wealth of information to attackers. Since we saved the results of the zone transfer to an output file, we can easily manipulate the results with UNIX programs like `grep`, `sed`, `awk`, or `perl`.

Suppose we are experts in SunOS or Solaris. We could programmatically find out the IP addresses that had an HINFO record associated with SPARC, Sun, or Solaris.

```
[bash]$ grep -i solaris zone_out |wc –l
     388
```

We can see that we have 388 potential records that reference the word "Solaris." Obviously, we have plenty of targets.

Suppose we wanted to find test systems, which happen to be a favorite choice for attackers. Why? Simple—they normally don't have many security features enabled, often have easily guessed passwords, and administrators tend not to notice or care who logs in to them. They're a perfect home for any interloper. Thus, we can search for test systems as follows:

```
[bash]$ grep -i test /tmp/zone_out |wc –l
     96
```

So we have approximately 96 entries in the zone file that contain the word "test." This should equate to a fair number of actual test systems. These are just a few simple examples. Most intruders will slice and dice this data to zero-in on specific system types with known vulnerabilities.

Keep a few points in mind. The aforementioned method only queries one nameserver at a time. This means that you would have to perform the same tasks for all nameservers that are authoritative for the target domain. In addition, we only queried the Acme.net domain. If there were subdomains, we would have to perform the same type of query for each subdomain (for example, greenhouse.Acme.net). Finally, you may receive a message stating that you can't list the domain or that the query was refused. This usually indicates that the server has been configured to disallow zone transfers from unauthorized users. Thus, you will not be able to perform a zone transfer from this server. However, if there are multiple DNS servers, you may be able to find one that will allow zone transfers.

Now that we have shown you the manual method, there are plenty of tools that speed the process, including, `host`, Sam Spade, `axfr`, and `dig`.

The `host` command comes with many flavors of UNIX. Some simple ways of using `host` are as follows:

```
host -l Acme.net
```

or

```
host -l -v -t any Acme.net
```

If you need just the IP addresses to feed into a shell script, you can just `cut` out the IP addresses from the `host` command:

```
host -l acme.net |cut

 -f 4 -d" " >> /tmp/ip_out
```

Not all footprinting functions must be performed through UNIX commands. A number of Windows products provide the same information, as shown in Figure 1-5.

Finally, you can use one of the best tools for performing zone transfers, `axfr` (http://ftp.cdit.edu.cn/pub/linux/www.trinux.org/src/netmap/axfr-0.5.2.tar.gz) by Gaius. This

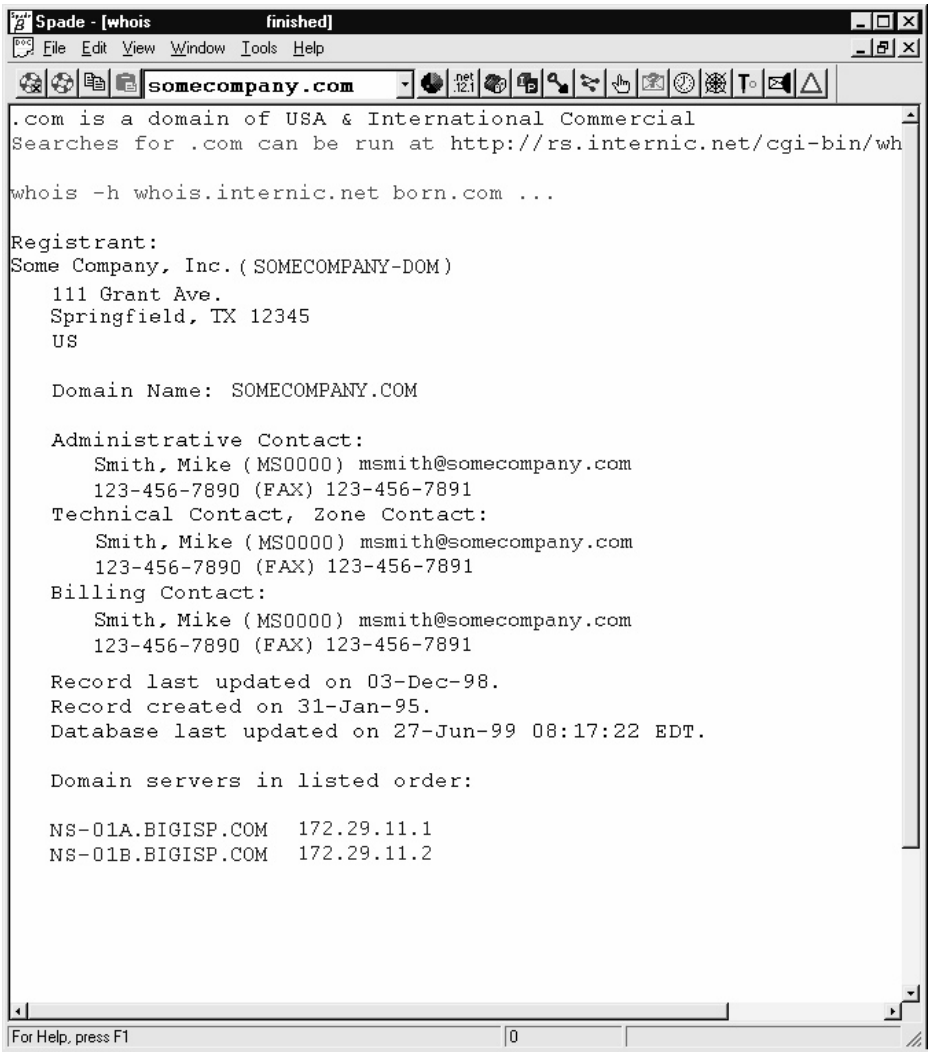

Figure 1-5. If you're Windows inclined, you could use the multifaceted Sam Spade to perform a zone transfer as well as other footprinting tasks.

utility will recursively transfer zone information and create a compressed database of zone and host files for each domain queried. In addition, you can even pass top-level domains like *com* and *edu* to get all the domains associated with `com` and `edu`, respectively. However, this is not recommended. To run `axfr`, you would type the following:

```
[bash]$ axfr Acme.net
axfr: Using default directory: /root/axfrdb
Found 2 name servers for domain 'Acme.net.':
Text deleted.
Received XXX answers (XXX records).
```

To query the `axfr` database for the information you just obtained, you would type the following:

```
[bash]$ axfrcat Acme.net
```

Determine Mail Exchange (MX) Records

Determining where mail is handled is a great starting place to locate the target organization's firewall network. Often in a commercial environment, mail is handled on the same system as the firewall, or at least on the same network. So we can use `host` to help harvest even more information.

```
[bash]$ host  Acme.net
Acme.net has address 10.10.10.1
Acme.net mail is handled (pri=20) by smtp-forward.Acme.net
Acme.net mail is handled (pri=10) by gate.Acme.net
```

If `host` is used without any parameters on just a domain name, it will try to resolve *A* records first, then *MX* records. The preceding information appears to cross-reference with the `whois` ARIN search we previously performed. Thus, we can feel comfortable that this is a network we should be investigating.

Countermeasure: DNS Security

DNS information provides a plethora of information to attackers, so it is important to reduce the amount of information available to the Internet. From a host configuration perspective, you should restrict zone transfers to only authorized servers. For modern versions of BIND, the *allow-transfer* directive in the *named.conf* file can be used to enforce the restriction. To restrict zone transfers in Microsoft's DNS, you can use the Notify option. (See http://support.microsoft.com/support/kb/articles/q193/8/37.asp for more information.) For other nameservers, you should consult the documentation to determine what steps are necessary to restrict or disable zone transfers.

On the network side, you could configure a firewall or packet-filtering router to deny all unauthorized inbound connections to TCP port 53. Since name lookup requests are UDP and zone transfer requests are TCP, this will effectively thwart a zone transfer attempt. However, this countermeasure is a violation of the RFC, which states that DNS

queries greater than 512 bytes will be sent via TCP. In most cases, DNS queries will easily fit within 512 bytes. A better solution would be to implement cryptographic Transaction Signatures (TSIGs) to allow only "trusted" hosts to transfer zone information. For a step-by-step example of how to implement TSIG security, see http://romana.ucd.ie/james/tsig.html.

Restricting zone transfers will increase the time necessary for attackers to probe for IP addresses and hostnames. However, since name lookups are still allowed, attackers could manually perform lookups against all IP addresses for a given net block. Therefore, configure external name servers to provide information only about systems directly connected to the Internet. External nameservers should never be configured to divulge internal network information. This may seem like a trivial point, but we have seen misconfigured nameservers that allowed us to pull back more than 16,000 internal IP addresses and associated hostnames. Finally, we discourage the use of HINFO records. As you will see in later chapters, you can identify the target system's operating system with fine precision. However, HINFO records make it that much easier to programmatically cull potentially vulnerable systems.

Step 4. Network Reconnaissance

Now that we have identified potential networks, we can attempt to determine their network topology as well as potential access paths into the network.

Tracerouting

Popularity:	*9*
Simplicity:	*9*
Impact:	*2*
Risk Rating:	***7***

To accomplish this task, we can use the `traceroute` (ftp://ftp.ee.lbl.gov/traceroute.tar.gz) program that comes with most flavors of UNIX and is provided in Windows NT. In Windows NT, it is spelled `tracert` due to the 8.3 legacy filename issues.

`Traceroute` is a diagnostic tool originally written by Van Jacobson that lets you view the route that an IP packet follows from one host to the next. `Traceroute` uses the time-to-live (TTL) option in the IP packet to elicit an ICMP TIME_EXCEEDED message from each router. Each router that handles the packet is required to decrement the TTL field. Thus, the TTL field effectively becomes a hop counter. We can use the functionality of `traceroute` to determine the exact path that our packets are taking. As mentioned previously, `traceroute` may allow you to discover the network topology employed by the target network, in addition to identifying access control devices (application-based firewall or packet-filtering routers) that may be filtering our traffic.

Let's look at an example:

```
[bash]$ traceroute Acme.net
traceroute to Acme.net (10.10.10.1), 30 hops max, 40 byte packets
1  gate2 (192.168.10.1)  5.391 ms  5.107 ms  5.559 ms
2  rtr1.bigisp.net (10.10.12.13) 33.374 ms 33.443 ms 33.137 ms
3  rtr2.bigisp.net (10.10.12.14) 35.100 ms 34.427 ms 34.813 ms
4  hssitrt.bigisp.net (10.11.31.14) 43.030 ms 43.941 ms 43.244 ms
5  gate.Acme.net (10.10.10.1)  43.803 ms  44.041 ms  47.835 ms
```

We can see the path of the packets leaving the router (gate) and traveling three hops (2–4) to the final destination. The packets go through the various hops without being blocked. From our earlier work, we know that the MX record for Acme.net points to gate.acme.net. Thus, we can assume this is a live host and that the hop before it (4) is the border router for the organization. Hop 4 could be a dedicated application-based firewall, or it could be a simple packet-filtering device—we are not sure yet. Generally, once you hit a live system on a network, the system before it is a device performing routing functions (for example, a router or a firewall).

This is a very simplistic example. But in a complex environment, there may be multiple routing paths, that is, routing devices with multiple interfaces (for example, a Cisco 7500 series router). Moreover, each interface may have different access control lists (ACLs) applied. In many cases, some interfaces will pass your `traceroute` requests, while others will deny it because of the ACL applied. Thus, it is important to map your entire network using `traceroute`. After you `traceroute` to multiple systems on the network, you can begin to create a network diagram that depicts the architecture of the Internet gateway and the location of devices that are providing access control functionality. We refer to this as an *access path diagram.*

It is important to note that most flavors of `traceroute` in UNIX default to sending User Datagram Protocol (UDP) packets, with the option of using Internet Control Messaging Protocol (ICMP) packets with the `-I` switch. In Windows NT, however, the default behavior is to use ICMP *echo request packets.* Thus, your mileage may vary using each tool if the site blocks UDP vs. ICMP and vice versa. Another interesting option of `traceroute` includes the `-g` option that allows the user to specify loose source routing. Thus, if you believe the target gateway will accept source-routed packets (which is a cardinal sin), you might try to enable this option with the appropriate hop pointers (see `man traceroute` in UNIX for more information).

There are several other switches that we need to discuss that may allow you to bypass access control devices during our probe. The `-p` *n* option of `traceroute` allows you to specify a starting UDP port number (*n*) that will be incremented by 1 when the probe is launched. Thus, we will not be able to use a fixed port number without some modification to `traceroute`. Luckily, Michael Schiffman has created a patch (http:// www.packetfactory .net/Projects/firewalk/traceroute.diff) that adds the `-S` switch to stop port incrementation for `traceroute` version 1.4a5 (ftp.cerias.purdue.edu/pub/tools/unix/netutils/traceroute/old/). This allows you to force every packet we send to have a fixed port number, in the hopes that the access control device will pass this traffic. A good starting port number

would be UDP port 53 (DNS queries). Since many sites allow inbound DNS queries, there is a high probability that the access control device will allow our probes through.

```
[bash]$ traceroute  10.10.10.2
traceroute to (10.10.10.2), 30 hops max, 40 byte packets
 1  gate (192.168.10.1)  11.993 ms  10.217 ms  9.023 ms
 2  rtr1.bigisp.net (10.10.12.13)37.442 ms  35.183 ms  38.202 ms
 3  rtr2.bigisp.net (10.10.12.14) 73.945 ms  36.336 ms  40.146 ms
 4  hssitrt.bigisp.net (10.11.31.14) 54.094 ms 66.162 ms  50.873 ms
 5  * * *
 6  * * *
```

We can see here that our `traceroute` probes, which by default send out UDP packets, were blocked by the firewall.

Now let's send a probe with a fixed port of UDP 53, DNS queries:

```
[bash]$ traceroute -S -p53 10.10.10.2
traceroute to (10.10.10.2), 30 hops max, 40 byte packets
 1  gate (192.168.10.1)    10.029 ms  10.027 ms  8.494 ms
 2  rtr1.bigisp.net (10.10.12.13) 36.673 ms 39.141 ms 37.872 ms
 3  rtr2.bigisp.net (10.10.12.14) 36.739 ms 39.516 ms 37.226 ms
 4  hssitrt.bigisp.net (10.11.31.14)47.352 ms 47.363 ms 45.914 ms
 5  10.10.10.2 (10.10.10.2)  50.449 ms  56.213 ms  65.627 ms
```

Because our packets are now acceptable to the access control devices (hop 4), they are happily passed. Thus, we can probe systems behind the access control device just by sending out probes with a destination port of UDP 53. Additionally, if you send a probe to a system that has UDP port 53 listening, you will not receive a normal ICMP unreachable message back. Thus, you will not see a host displayed when the packet reaches its ultimate destination.

Most of what we have done up to this point with `traceroute` has been command-line oriented. For the graphically inclined, you can use VisualRoute (http://www.visualroute.com) or NeoTrace (http://www.neotrace.com/) to perform your tracerouting. VisualRoute provides a graphical depiction of each network hop and integrates this with `whois` queries. VisualRoute, depicted in Figure 1-6, is appealing to the eye, but does not scale well for large-scale network reconnaissance.

There are additional techniques that will allow you to determine specific ACLs that are in place for a given access control device. *Firewall protocol scanning* is one such technique and is covered in Chapter 11.

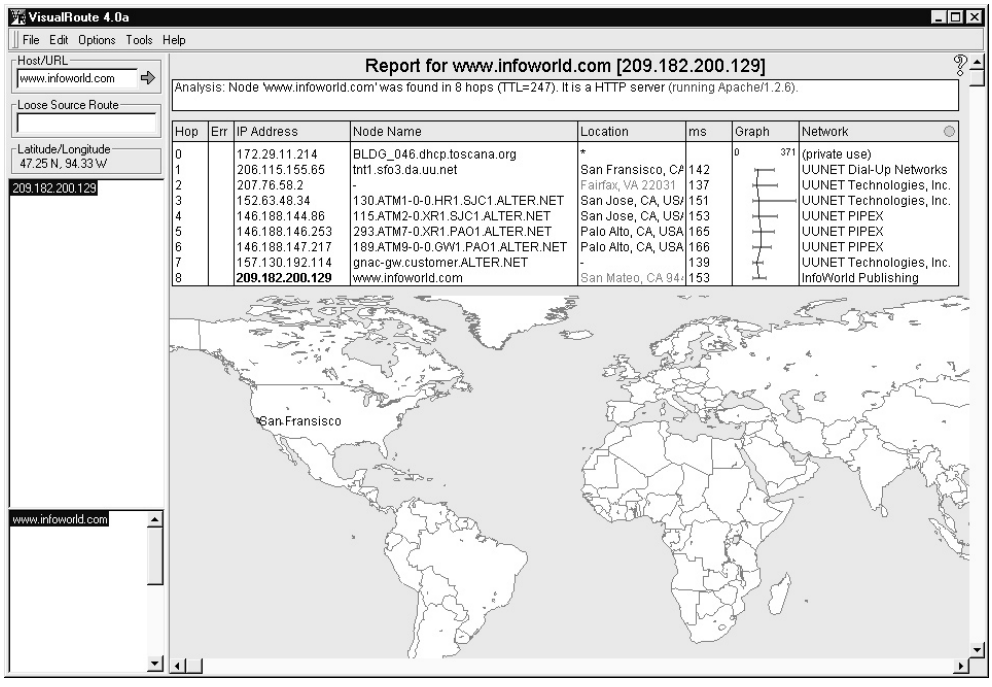

Figure 1-6. VisualRoute, the Cadillac of traceroute tools, provides not just router hop information but also geographic location, whois lookups, and web server banner information.

Countermeasure: Thwarting Network Reconnaissance

In this chapter, we only touched upon network reconnaissance techniques. We shall see more intrusive techniques in the following chapters. There are, however, several countermeasures that can be employed to thwart and identify the network reconnaissance probes discussed thus far. Many of the commercial network intrusion detection systems (NIDSes) will detect this type of network reconnaissance. In addition, one of the best free NIDS programs, snort (http://www.snort.org/) by Marty Roesch, can detect this activity. If you are interested in taking the offensive when someone traceroutes to you, Humble from Rhino9 developed a program called RotoRouter (http://packetstorm.securify.com/UNIX/loggers/rr-1.0.tgz). This utility is used to log incoming `traceroute` requests and generate fake

responses. Finally, depending on your site's security paradigm, you may be able to configure your border routers to limit ICMP and UDP traffic to specific systems, thus minimizing your exposure.

SUMMARY

As you have seen, attackers can perform network reconnaissance or footprint your network in many different ways. We have purposely limited our discussion to common tools and techniques. Bear in mind, however, that new tools are released daily. Moreover, we chose a simplistic example to illustrate the concepts of footprinting. Often you will be faced with a daunting task of trying to identify and footprint tens or hundreds of domains. Therefore, we prefer to automate as many tasks as possible via a combination of shell and `expect` scripts or `perl` programs. In addition, there are many attackers well schooled in performing network reconnaissance activities without ever being discovered, and they are suitably equipped. Thus, it is important to remember to minimize the amount and types of information leaked by your Internet presence and to implement vigilant monitoring.