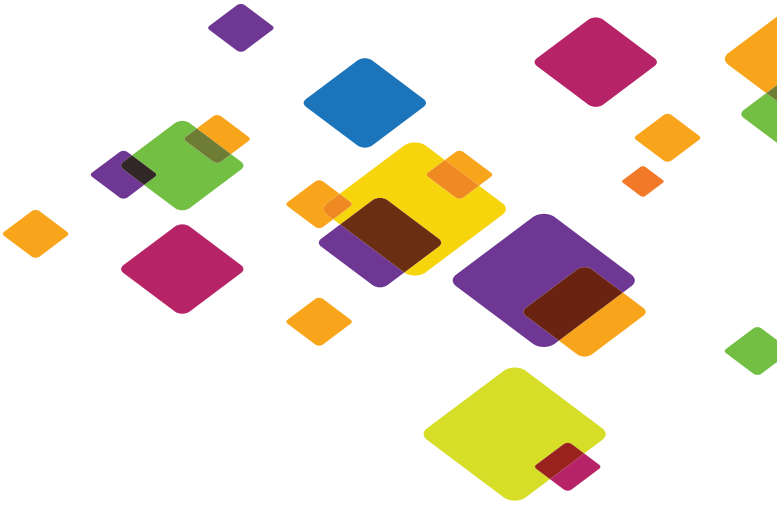
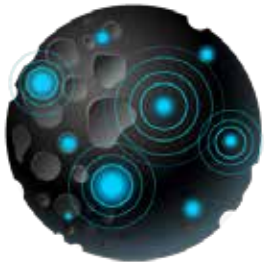
Infoblox
CONTROL YOUR NETWORK

WHITE PAPER

A Cybercriminal's Guide to Exploiting DNS for Fun and Profit





Without DNS, the Internet Stops Working Like It Should

And that's great news for people like you.

Most people with online connections are happy using the Internet for "traditional" applications: sending email; watching videos; buying books and clothes; or keeping up with friends and family. But for you – one of an elite group of cybercriminals, "black hat" hackers and other Internet super-users – the Web has much, much more to offer. For you, the Internet is nothing less than a vast collection of vulnerabilities waiting to be tapped.

That is why enterprising Web cybercriminals need to understand the Domain Name System, or DNS. DNS represents the absolute cutting edge in what might be politely described as "Web vulnerability entrepreneurship." It's where the best and the brightest, not to mention the best and the baddest, are spending their time.

While other forms of vulnerability utilization have been declining due to increased "awareness" by security software providers and others, DNS-based exploits are on the rise, up 200 percent from 2012 to 2013, according to one study, faster than any other category. DNS is now the second-most-common vector of Internet exploits, behind only the venerable HTTP.

Who should learn about DNS? Just about anyone interested in either launching some form of Denial of Service attack, or else redirecting Web traffic away from its proper server to one that you control. This includes you, if:

- You're a pragmatic and results-oriented businessperson seeking to quickly gain a competitive edge in the marketplace by slowing down, or shutting altogether, the Web site of your competitor(s).
- You're a member of a group like Anonymous, and you're anxious to battle an injustice you just read about on an IRC channel by disabling the Web sites of the businesses or institutions mentioned in the post.
- You've just been fired, and you're anxious to show your former employer how wrong they were by "causing a few problems" for their Web site.
- You're looking to supplement your income by taking advantage of the growing opportunities to charge companies thousands of dollars to put their Web sites back online.

The following overview of how to use DNS to your advantage is a high-level one that, for the sake of brevity, omits many technical details. But if you're confused about something, or need precise, step-by-step instructions about how to perform some specific DNS attack, don't worry. There are plenty of more detailed how-to guides online, as well as helpful chat rooms and other community sites, easily findable via just about any search engine.

Note from Infoblox: While we hope it's obvious, this paper is meant to be a humorous treatment of a worrisome reality affecting every business with online operations: The increased use of DNS attacks by cybercriminals. Infoblox, of course, does not support or encourage cybercrime; to the contrary, our company is working to help companies defend themselves against DNS-based threats. This white paper contains no information on DNS exploits that hasn't been widely covered in both the mainstream and trade press, and while we believe all of the technical information to be accurate, there isn't nearly enough information in this paper to assist a real criminal in actually mounting any of the DNS attacks we describe.

DNS is the Internet's "Directory Assistance"

And how it can be of direct assistance to you.

Remember "directory assistance" for the telephone system, when it was still centrally controlled by the phone company? Think of all the havoc you could have wreaked if you seized control of it. You could, for example, have given out a fake phone number – one that rang on your desk – to anyone who wanted the number of a competitor. Or, you could tell all the 411 operators to give the same number to everybody, regardless of what number they asked for. That phone would be getting so many "wrong number" calls it would effectively be out of commission.

That's essentially what you are able to do with the Domain Name System. It's the job of DNS to provide the numerical Internet Protocol (IP) address of a server to someone who only knows the server's domain name; it's the Web equivalent of looking up a phone number. But the Internet is much too big for a single DNS server to meet the needs of all users. Instead, DNS functions are performed by a hierarchy of hundreds of thousands of special computers all around the world. The machines operate in the decentralized, collaborative fashion that is common on the Internet. If one DNS server doesn't know the answer to a query – in other words, if it doesn't know the IP address associated with a particular Web address – it asks another DNS server, using established protocols.

There are many different kinds of DNS servers; some will answer queries from just about anyone; others will only talk to a small number of other machines. Some of these servers are high-end devices operated by elite Internet Service Providers (ISPs). But many thousands of others are run by businesses in connection with their Web operations. And, as with everything else on the Web, some of these servers are going to be less carefully maintained than others, and thus much more easily exploitable by you and your friends.

One of the most exciting aspects of DNS is that you don't need to concern yourself with the security details of the Web site that is your actual target. With some DNS exploits, you can deal with other servers that your target has no connection with, and likely doesn't even know about. Your local bank might have the best-protected Web site on the entire Internet. But if you are able to get a DNS server somewhere else to direct the bank's "rightful" traffic to you, all of the bank's security features will be for naught! Even better, because the exploit is occurring elsewhere on the Internet, on servers your bank doesn't control, tracking down and then stopping your attack becomes an arduous, time-consuming task. In many cases, you'll be able to have your way with the users trying to get to your targeted servers for days, even weeks!

Let's Get Real!

Some actual DNS attacks, ripped from the headlines.

At this point, you may be saying to yourself, "All this sounds good in theory. But does it really work?" The answer is a resounding yes! Consider these recent DNS-related adventures undertaken by people not much different than yourself.

- Google saw one of its home pages in the Middle East changed not because one of its servers was compromised, but instead via clever manipulation of DNS. Visitors to the page designed for Google users in the Palestinian territories saw a number of irreverent messages, among them, "Listen to Rihanna and be cool."

- Both the New York Times and the Washington Post have been taken down via DNS in recent months. The New York Times case shows how easy it can be. The newspaper was using a commercial DNS servicer in Australia called Melbourne IT. That company, in turn, has hundreds of resellers. Someone gained access to the username and password of one of those resellers, logged in to Melbourne IT's system, and then changed the Times' DNS information. In the case of both newspapers, a group called the "Syrian Electronic Army" claimed responsibility. But no one really knows if the "Army" exists, or if it is just a single user no different from you.

- Spamhaus is an international organization responsible for maintaining a list of mail servers known to be associated with spam attacks. Virtually every form of anti-spam technology makes use of Spamhaus's database, often on an hourly basis. Spamhaus was taken offline for several days by a DNS-based Denial of Service episode using a technique called "DNS amplification." Because of the attack, the servers at Spamhaus were suddenly forced to cope with as much as 300 gigabits per second of data – well beyond what they, or just about any other set of servers, were architected for or can handle. One would think that creating such an enormous amount of data would be difficult, requiring, say, a botnet containing thousands of computers. But thanks to the basic design of the DNS system, it can be trivially easy – as we will see below.

- The same techniques that were used against Spamhaus are also being used against banks and financial institutions. Many of the biggest banks in the United States have found their Web sites unavailable because their DNS servers have been flooded with data; in some cases, the attacks were timed to coincide with efforts to transfer money out of the banks, as happened with San Francisco-based Bank of the West in 2013, to the tune of nearly $1 million. These institutions rarely like to talk about these issues, but we know the problem is severe. The U.S. Department of Justice recently granted a special antitrust waiver to all top U.S. financial institutions so they could come together to share technical information, in the interest of helping the FBI halt DNS attacks.

In The Belly of the Beast

Selected DNS attacks, up close and personal.

There are dozens of DNS attacks to choose from, with new ones being added all the time and traditional ones improved to preserve or increase their effectiveness. The list is far too long to be covered in detail in one place, but here are overviews of the most common. Remember, all these accomplish one of two things: They either re-route “legitimate” Internet traffic to servers that you control, or they flood servers with so much traffic that the Web sites of your targets are effectively taken off-line. What you do next is limited only by your entrepreneurial imagination.

Cache Poisoning

Cache poisoning is the functional equivalent of getting a directory assistance operator to give out phone numbers that you have selected, rather than the proper ones. This is one of the most popular DNS attacks, and the technique fully lives up to its scary-sounding name. And, as with other DNS exploits, new methods for cache poisoning are being invented all the time.

Remember that the job of a DNS server is to match a Web site name with the IP address for a specific server. With cache poisoning, you simply substitute the real IP address of a Web site with an IP address of your choosing – presumably one for a server you control.

You can design the “fake” page to be anything you want it to be. It can convey a message that you think is important, or, with a little bit of extra work, it can look exactly like the home page for, say, a bank -- to the point of asking for a username and password, which, of course, you will promptly collect.

Cache poisoning is a good example of how DNS exploits have evolved over time to escape the detection systems that companies are gradually putting in place, and thus make the exploits even more powerful.

The exploits take advantage of the fact that all DNS-IP pairs are temporary; the pairing of a certain IP server address to a particular domain might be true for only a few minutes, at which point a new pairing will take effect. (This is called a domain name’s “Time To Live” or TTL; Web sites use TTLs to balance their traffic throughout the day, moving visitors from one server bank to another as demand ebbs and flows.) This means every DNS server will periodically be asking other DNS servers for information, even for the most common sites.

DNS protocols allow an attacking server aiming to poison the cache of another server to ask how much time is left on a particular domain name-IP pair. Then, at the appointed second, the attacking server sends updates to the server containing the new, phony IP address.

There is a certain amount of luck involved in getting the transaction right. For example, legitimate DNS requests generate a 16-bit “Message ID” when sending out a query. When the server responds, it needs to include the same number. That initially proved difficult for people attempting cache poisoning exploits, until it was discovered that the random number generator creating the Message IDs had a bug that made the numbers much easier to predict.

A successful cache poisoning exploit accomplishes two things. It changes the domain-IP pair inside the machine so users are directed to a Web site controlled by the person running the exploit. In addition, it resets the domain name's Time To Live, sometimes from a few minutes to many years, to keep the bogus information intact for as long as possible.

It's important to appreciate that "cache poisoning" is not a single exploit, but a family of maneuvers that has changed over time, with new techniques constantly being developed. Some are far simpler than others both to plan and to defend against. One of the newer methods, called the "Kaminsky vulnerability," is fiendishly difficult to defend against, and while it was discovered in 2008, some DNS servers remain unprotected from it.

DNS Amplification and Reflection

As with anything in life, a Denial of Service attack can be done the hard way or the easy way. The hard way might be to painstakingly assemble a huge botnet of "infected" computers, and then have each machine send some sort of traffic to the targeted website.

The easy way is to let DNS do the work for you. This is what happens with DNS amplification and reflection.

In a "normal" DNS request, users tell a DNS server two things: the name of the web server they need an address for, and the IP address to send the information to – which, of course, would normally be the user's IP address. But in a DNS Amplification exploit, changes are made to both of those items. The changes are small, but the results can be dramatic.

DNS protocols allow for many different types of queries to be made of a DNS server. By far the most widely-used of these queries is an "A" query, when you want to know the IP address associated with a Web name. That is usually only a few bytes worth of information. But another query is "ANY," which tells the DNS server to report back all information it has about a domain name; for example, its cryptographic keys and security signatures. While ANY queries themselves are tiny, typically less than 50 bytes, the data payloads they trigger in response can be relatively enormous, more than 4,000 bytes.

Use (or actually "misuse") of the ANY request is key to DNS Amplification attacks. You simply create an ANY query that asks for the reply to be sent not to the original computer making the request, but instead to the Web server where you want service denied. This, fortunately, is as easy as mailing an envelope with a fake return address; the DNS protocol does nothing to verify this "return address" before replying.

So, you've tricked the DNS server into sending out vastly more information per request than it usually does, and you've also tricked it into sending all that data to another Web site. A little math should how devastating the combination can be. A computer with a 2 megabit per second (Mbps) outbound connection - most home cable broadband customers have that amount of bandwidth these days – can send out nearly 6,000 DNS queries a second, generating a tsunami of 200 Mbps at the targeted computer. With just five such machines, you're at a gigbit of data. Imagine what an entire botnet could do. Which is one of the reasons Distributed Denial of Service (DDoS) attacks measuring in the hundreds of gigabits a second have become common. Thanks, DNS!

And more!

Remember, these are just a few of the better-known DNS vulnerabilities. There are many more, with names like “tunneling,” “hijacking,” and “flood.” And new ones are being discovered all the time. Their internal mechanisms might be different, but they all have the goal: To take a crucial but fragile part of the Internet and make it work for evil instead of good.

There is something else working in your favor as a budding cybercriminal: IT inertia. Some of the biggest targets for DNS attacks are unpatched servers. That’s because, as with anything involving computers, DNS servers require a substantial body of software – one popular name server program consists of around 50,000 lines of code. While relatively compact compared to the millions of lines of code in a PC operating system, it’s still complex enough that bugs and glitches will inevitably appear, especially since the DNS system has been built up incrementally over several decades.

In just one example, a researcher recently discovered that a standard DNS server wasn’t properly performing a randomization function necessary to protect the integrity of a DNS request. The glitch greatly simplified the task of filling the server with bad data, and received extensive publicity in the IT community. But studies showed that long after the bug was first discovered, up to a quarter of all DNS machines had not been patched.

That gave cybercriminals like yourself plenty of time to launch attacks. And while that particular window of vulnerability will eventually be closed, be patient. A new one will open soon enough.

A Final Note

It should be obvious by now that DNS attacks are relatively simple to carry out, and have the potential for giving you superhero powers to disrupt Web sites that you’ve targeted. DNS attacks are a rapidly evolving field, and approaches that are popular today might not be effective tomorrow. But that’s no excuse not to get started. It’s always possible that effective “security” products will drastically reduce, or even eliminate, DNS attacks. But until that occurs, you should realize that we are living in something of a Golden Age for DNS Attacks. Isn’t it time you started living the dream?

CORPORATE HEADQUARTERS:

+1.408.986.4000

+1.866.463.6256

(toll-free, U.S. and Canada)

info@infoblox.com

www.infoblox.com

EMEA HEADQUARTERS:

+32.3.259.04.30

info-emea@infoblox.com

APAC HEADQUARTERS:

+852.3793.3428

sales-apac@infoblox.com