

SPF Overview

Meng Weng Wong

(Reprinted from Linux Journal)

You can help eliminate the spam problem by making it easy to detect forgeries. Protect your e-mail address reputation with a simple DNS technique.

SPF is an emerging antiforgery standard that aims to prevent worms, viruses and spam from forging arbitrary e-mail addresses as the envelope sender in SMTP. SPF has two parts: domain administrators need to publish SPF records in the DNS, and e-mail administrators need to install SPF-enabled MTAs to read those records. SPF records indicate the servers from which a domain sends outbound mail. Mail coming from anywhere else is considered forged.

This article, the first of a two-part series, explains the concepts and trade-offs involved in SPF protection and shows DNS administrators how to set up SPF records. The second article is aimed at showing e-mail administrators how to activate SPF protection in their MTAs. This article was written in early January 2004 and reflects the state of the Internet current at that time.

Worms, Viruses, Joe-Jobs and Envelope Sender Forgery

I got spam from myself today. I founded pobox.com, and I'm an e-mail guy. So I pressed H for headers and read the Received lines. Just as I thought: like much of the spam I receive, this one came from a broadband machine. It's probably an old PIII running Windows 2000 unpatched, used for gaming and MP3s, quietly humming at the foot of someone's bed, draped in dirty underwear. Maybe it lives on a potato farm in Idaho; maybe it looks out over Central Park. Either way, it's probably infected with a variant of the Sobig virus, written under contract to a spammer. The machine's rightful owner has no idea he's infected, no idea his machine has been sending a few hundred spams and viruses every hour since that forgotten day long ago when he clicked on that weird attachment that didn't open.

Spam messages disguise their origins. Spammers use compromised machines to send the spam. They forge message headers. They fake Received headers to throw off the scent, make up bogus Subjects to trick Bayesian filters and forge From lines pretending to be PayPal or eBay.

Spammers also forge the return path. When messages are undeliverable, they bounce back to the sender whose address is in the return path. Not the From: address in the message headers, but the return path of the SMTP envelope, the RFC2821 MAIL FROM. Often, spamware uses lists of old addresses, or they simply guess common user names or launch a dictionary attack. The result is a lot of bad addresses and a lot of bounces.

Spammers don't want those bounces. They'd rather somebody else receive them. So, they pick an address at random or use the recipient's address. That's how they made it look like I got spam from myself. Sometimes they choose a hated enemy and maliciously forge his address so he gets flooded with thousands of bounces.

In 1997, a spammer forged a return address at joes.com, which then was flooded by so many bounce messages it went down for ten days—and gave the world the term joe-job. Hotmail and AOL get joe-jobbed every day: a lot of spam pretends to be from AOL but doesn't really come through their servers. Under conventional SMTP, AOL can't do anything about it. If you put the AOL logo on a T-shirt and tried to sell that shirt, AOL's lawyers would have you ceasing-and-desisting in a heartbeat. But spammers forge @aol.com every day. They can get away with it because they use SMTP.

The Simple Mail Transfer Protocol (SMTP) was designed more than 20 years ago—a kindlier, gentler time. The entire Internet was only a handful of research institutions. SMTP has served us well since then, but it's beginning to show its age.

SPF Overview

Meng Weng Wong

(Reprinted from Linux Journal)

SMTP is open and trusting. Its rules are relatively lax. You can assert any envelope sender and make up all the headers you want. You could argue today, though, that a protocol that lets joe-jobs happen is a little too open, a little too trusting. That's where sender authentication comes in. SPF tightens the rules.

Sender Authentication with SPF

When you send mail to a domain, your MTA does a DNS lookup (an MX query) to find out to which server to route the mail. Such a server is called a mail exchanger (MX). Small domains tend to have only one MX server. Big domains tend to have more. Mail to a domain goes to its MX servers.

Now for the big idea. In 99% of all cases, when a domain sends mail, that mail originates from a relatively small set of servers controlled by that domain. The domain could designate those servers using the DNS, then announce that any mail not received from those servers probably is forged. That's called a designated sender scheme (Figure 1).

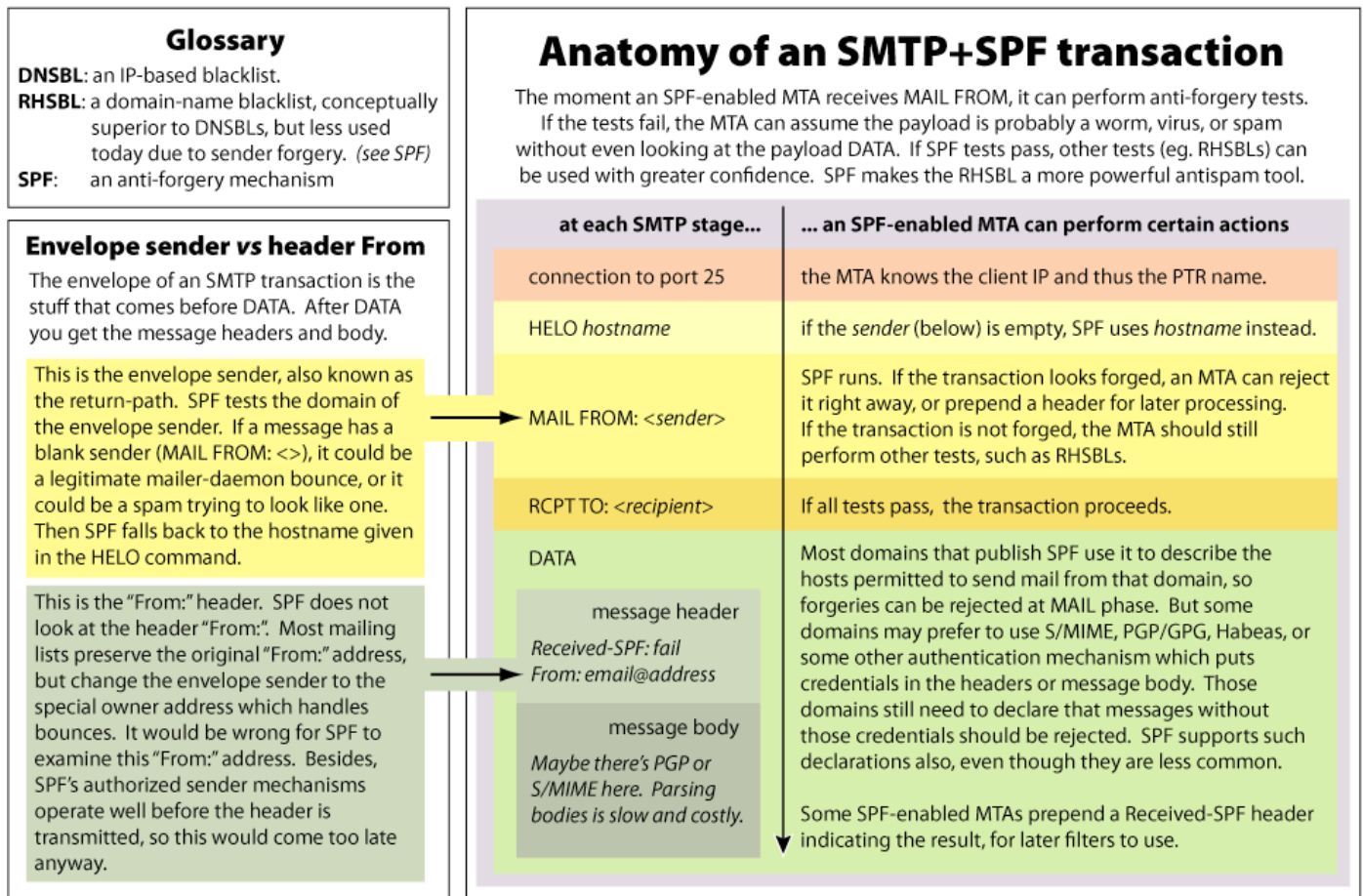


Figure 1

With SPF, one mail server can check whether another server really is associated with the address the mail claims to be from.

SPF Overview

Meng Weng Wong

(Reprinted from Linux Journal)

Designated sender schemes are useful because they help fight forgery and are easy to set up. After all, domain owners already know which servers send mail from that domain. When I say send mail from that domain, I mean originate an SMTP transaction where the MAIL-FROM envelope sender shows that domain. I'm not talking about the From: header. This is an important distinction.

Mail from a domain tends to come from a small number of servers. That's true for domains large and small. Mail from aol.com comes from AOL's servers. Mail from my personal domain comes from my personal servers. It certainly doesn't come from a machine covered in dirty underwear.

Many ISPs already are implementing these kinds of rules in a haphazard and often slightly broken way. The problem is, one ISP doesn't know the insides of another ISP, and it's easy to guess wrong. Maybe aol.com's mail servers also originate mail for aol.net or vice versa. Wouldn't it be better if AOL themselves announced their designated servers in a simple, flexible, extensible, open format that everybody could use?

Well, they do. SPF is a standard, flexible, extensible, open format that everybody can use. At the time of this writing, AOL recently had started publishing their SPF record.

MTAs can interpret that record and use it to tell whether mail that claims to be from @aol.com is a fake.

SPF Overview

Meng Weng Wong
(Reprinted from Linux Journal)

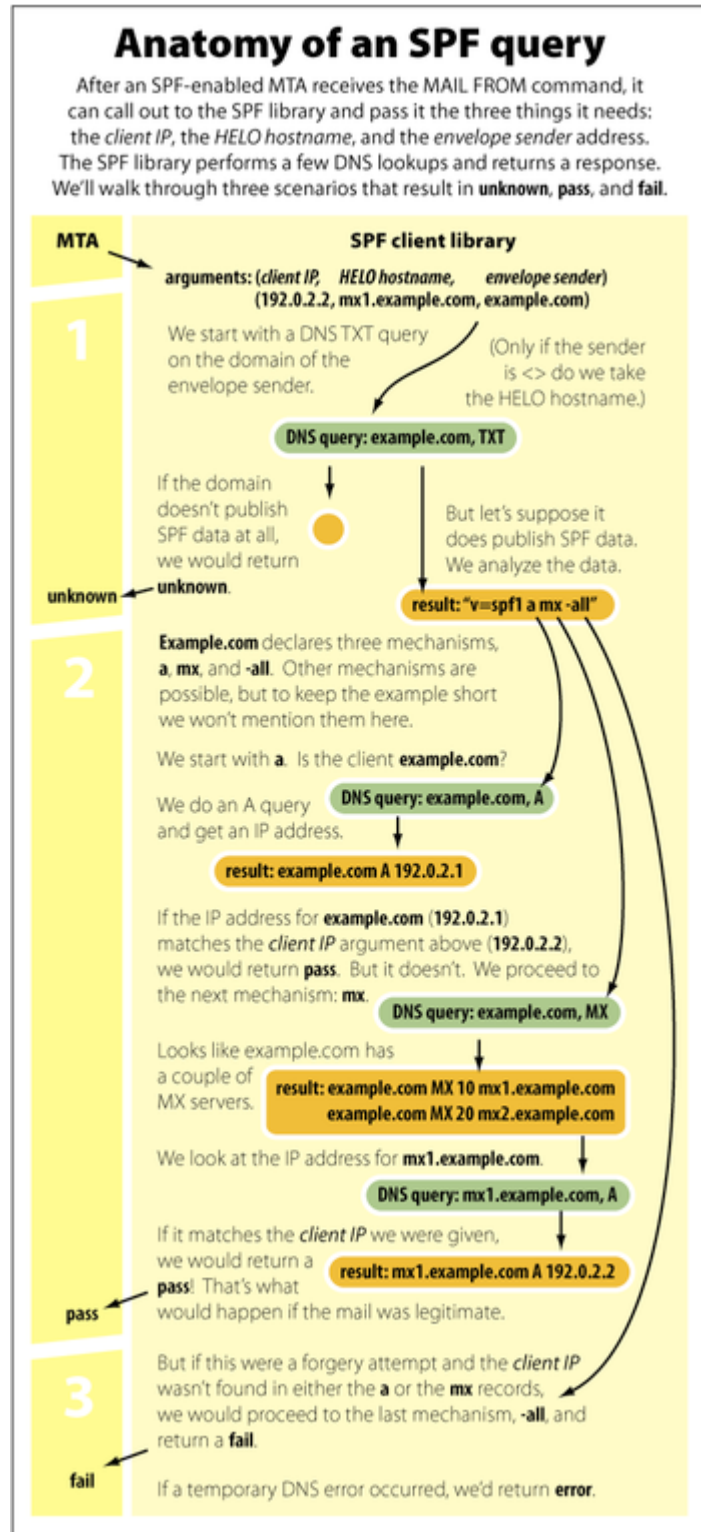


Figure 2
SPF performs a simple DNS-based lookup for each incoming message.

SPF Overview

Meng Weng Wong

(Reprinted from Linux Journal)

All this rule tightening is purely voluntary: domains that don't publish SPF records can continue to send mail as before. Some unusual domains might be served better by not publishing SPF; it's up to them. But most domains should want to use SPF.

To publish SPF, a domain has to add only one line to its zone file. That line is a TXT record, and you can publish it today. Let's see what the TXT record looks like.

SPF by Example

Suppose example.com wants to publish SPF. It expects MTAs everywhere to read its SPF record and use it to reject forgery attempts. It hopes SPF reduces the volume of joe-job bounces and bogus abuse reports. So it adds the following line to its zone file:

```
example.com. IN TXT "v=spf1 a mx ptr -all"
```

The v=spf1 version string identifies this as an SPF record. The -all means reject all mail by default. Domains that don't send any mail, such as altavista.com, can get by with simply v=spf1 -all. But if the domain does send mail, it declares mechanisms that describe how legitimate mail should look. Mechanisms go in the middle, before -all. The first mechanism to match provides a result for the SPF query. -all always matches and so belongs at the end.

Basic SPF

A: the A mechanism means the IP address of example.com is permitted to send mail from example.com. If you want to say the IP address of some-other.com is permitted, you can say a:some-other.com. You can use as many A mechanisms as you want.

MX: the MX mechanism means the MX servers for example.com all are permitted to send mail from example.com. If you want to say the MX servers for some-other.com are permitted, you can say mx:some-other.com. You can use as many MX mechanisms as you want.

PTR: the PTR mechanism says if a host has a PTR record that ends in example.com, it is permitted to send mail from example.com. This would be a good choice for Yahoo, whose mail server names all end in yahoo.com. It would be a bad choice for a broadband provider like Comcast. If you want to say servers whose names end in some-other.com are permitted to send mail from example.com, you can say ptr:some-other.com. You can use as many PTR mechanisms as you want.

IP4: to say the class C network of 192.0.2.0 is permitted to send mail from example.com, you would write ip4:192.0.2.0/24.

Mechanisms are interpreted left-to-right. Using v=spf1 a mx ptr -all first would check whether the connecting client was found in the A record for the domain or, failing that, in its list of MX servers. Then the MTA would check to see whether the hostname of the client matched the domain. If none of the mechanisms matched, -all would be evaluated, the result would be fail and the MTA would be justified in rejecting the mail.

A, MX, PTR and IP4 are enough for the overwhelming majority of domains. The setup wizard at spf.pobox.com/wizard.html can help you configure SPF for your domain. But if your situation is complex, you can use the mechanisms described in the "Advanced SPF" sidebar.

SPF Overview

Meng Weng Wong

(Reprinted from Linux Journal)

Extensibility

SPF has a number of built-in mechanisms. The basic ones let you designate the hosts that send mail from your domain. This works well for almost all domains out there, because each domain's mail comes only from a small set of hosts. But if mail from your domain is distinguished in some other way, say you always sign it with S/MIME, instead of typing a or mx you can type smime.

Using designated sender mechanisms (A, MX, PTR and IP4) is one possible approach to sender authentication. New sender authentication methods are being developed. SPF is extensible, though, so it can work gracefully with them. SPF plugins that understand future extension mechanisms will be able to interpret them correctly. SPF plugins that don't understand those mechanisms will return unknown, and your domain will be treated as though it did not have an SPF record at all.

Protecting Subdomains and MX Servers

Today, spammers forge domain names. Tomorrow, they might forge hostnames. They might try to joe-job your laptop by making up username@ibook.example.com. It's a good idea to protect your subdomains as well. You should start with your MX servers and move on to other hosts with A records. Here's why.

Bounce messages are sent with MAIL FROM: <>. The null sender address ensures that bounces don't themselves bounce and create a loop. When SPF sees the null sender address, it falls back to the hostname given in the HELO command. When your MTA sends a bounce message, it announces its hostname in the HELO command it sends. If that hostname has an SPF A mechanism listed, the message passes. So SPF prevents HELO forgery as well.

Traveling Mailman and the Forwarding Problem

SPF was designed to give the greatest benefit for the least cost. It tightens the rules in a way that makes it hard for bad people to do bad things, while not bothering the good people who do good things. Even so, some power users who have taken advantage of SMTP's lax rules may be inconvenienced by SPF. This section describes the two problems SPF causes power users and offers ways to work around them.

Most end users relay their outbound mail through their ISPs' SMTP servers. Most modern clients also support SASL authentication or POP-before-SMTP for users who need to phone home from outside the ISPs' networks. Users who always send mail through their ISPs' SMTP servers are automatically SPF-compliant and don't need to do a thing.

But some power users with an MTA on their laptop are used to originating mail from random IP addresses, bypassing their ISPs' SMTP servers entirely. SPF accommodates these users: the advanced mechanism (see the "Advanced SPF" sidebar) is a way to exempt certain users from being required to use their ISPs' SMTP servers. They can keep doing what they want.

Advanced SPF

Exists: the Exists mechanism takes an argument that expands to a domain name, and you can use macros. For example, exists:%{ir}._spf.example.com might expand to 2.2.0.192.ceo._spf.example.com. An SPF client would perform an A query on the expanded domain name, and if it got back any A record (for example, 127.0.0.2) Exists would result in a pass. You could use this technique to allow the special user ceo@example.com to send mail from a particular host, say 192.0.2.2, by creating an A record corresponding to the domain name above. Some people have written custom DNS servers to handle complex Exists queries. With Exists, the sky's the limit.

SPF Overview

Meng Weng Wong

(Reprinted from Linux Journal)

Include: if you send mail through another organization's servers, you should use the Include mechanism to point to their domain, so the SPF record is pulled in and expanded. For example, a vanity domain might use include:isp.com if it sends mail through ISP.com's mail servers. Any server permitted to send mail for ISP.com then is permitted to send mail for the vanity domain. You can include multiple other domains.

The modifiers Redirect and Exp: modifiers are different from the other mechanisms we've seen so far; they use equal signs instead of colons. Although mechanisms can repeat, you can have only one modifier per SPF record. Redirect is a modifier that works like Include, except the original query is replaced completely by the new query. Exp lets you define an explanation string. If an MTA rejects a forgery attempt, the explanation string appears in the SMTP error message that goes back to the original sender. You may have legitimate users who aren't using your SMTP servers, and SPF quickly can find out who they are. You also can set the explanation string to a URL that points to further information on how to configure mail clients correctly. All these mechanisms are described in detail at spf.pobox.com/mechanisms.html.

Some power users have a dozen or more addresses that forward all over the place by using entries in /etc/aliases or .forward files. In classical forwarding, the envelope sender remains unchanged while the recipient address is rewritten. This becomes a problem, though, when the message arrives at the destination—it still has the original sender address, and SPF tests fail.

The workaround is easy, however; you simply need to switch to remailing, where the sender address changes as well. There are many ways to accomplish this. Read the SPF FAQ (spf.pobox.com/faq.html#forwarding) to pick up the one that's right for you. Most end users have nothing to do with forwarding; only power users need to implement this workaround. If you have third-party service through an alumni, vanity domain or other commercial forwarding provider (such as pobox.com), you should expect them to implement remailing for you.

Stopping Spam: It's Part of the Solution

The primary goal of SPF is to stop forgery. I don't want to get any more spam from myself, and I certainly don't want you to receive any spam that claims to be from me. Worms and viruses tend to forge the envelope sender, too, and we can block them with SPF. And, stopping forgery carries a bonus. When spammers are forced to use their true names, we can figure out which domains are legitimate and which are spammers. People already are doing this: a right-hand side block list (RHSBL) is the domain name version of a DNS block list (DNSBL). Spammers who aren't afraid of using their own domains end up on RHSBLs quickly, and they can be blocked that way. In an SPF world, RHSBLs will become more important and effective.

Why Do People Use SPF?

Big domains, including ISPs, banks and well-known brands care about controlling their trademarks. They have an obligation to protect their names. Altavista.com publishes an SPF record as do AOL and Oxford. More domains get on the bandwagon every day. Smaller domains publish SPF's simply because they don't want to be joe-jobbed.

On the receiving end, ISPs upgrade their MTAs and turn on SPF simply because it means less forgery—less spam, worms and viruses. Their bandwidth costs go down, too, because SPF lets them cut off the spammer before data is transmitted. They don't have to perform any cryptography or verify any signatures. SPF saves money.

SPF Overview

Meng Weng Wong

(Reprinted from Linux Journal)

Adoption

By the time this article is published, SPF support should be either bundled in or available as a downloadable plugin for the latest versions of SpamAssassin, Postfix, Sendmail, Exim and qmail. Commercial antispam vendors have committed to support SPF; Declude JunkMail, for one, reports that SPF is successfully blocking spam in the field.

If all goes well, the SPF standard will be published as an RFC in the near future. But thousands of domains, including some quite large ones, already publish SPF records. There's no reason to wait; you should publish SPF today.

SPF and Conventional Antispam Methods

DNS blacklists or blocklists (DNSBLs): IPv4 space is 32 bits wide; 2³² is about 4.2 billion—4.2 billion grains of sand would just about fill a pickup truck. Imagine trying to paint each individual grain black or white. IP-based blacklists are a valiant effort, but they operate at too low a level. A good DNSBL has to decide whether an IP address is spammy and get it right for each of the 4.2 billion IP addresses. No wonder DNSBLs come and go—their maintainers burn out and give up.

Right-hand side blacklists: RHSBLs use domain names, whereas DNSBLs use IP addresses. Domain names are a much better way to identify entities on the Internet, but RHSBLs haven't been quite as popular as DNSBLs. Why not? Spam doesn't come from spammer.net. It's forged from yahoo.com. That's why SPF helps: if spammers send mail with their true names, blocking them becomes trivial.

Address verification: at MAIL phase, you can check the validity of the envelope sender by attempting to send a test message to it. If the test comes back user unknown, you might not want to accept the message. This is useful because spammers often make up addresses at random. But as address verification becomes more common, spammers can be expected to forge actual addresses—all the more reason to use SPF.

Signature solutions: PGP/GPG and S/MIME users sign their messages. Recipients can check signature validity by downloading keys from a key server. Other schemes have been proposed in which the DNS itself acts as the repository for public keys. These solutions are good because .forward files continue to work without modification. They are bad, however, because a message has to cross the pipe, costing bandwidth and CPU, before its legitimacy can be determined. In any case, a domain that uses these mechanisms still can use SPF to announce that any messages without a signature should be rejected.

Challenge/response: you don't want to send challenges to spam, especially not forged spam. If SPF tells you a sender address definitely was forged, you can junk the message without bothering to challenge it.