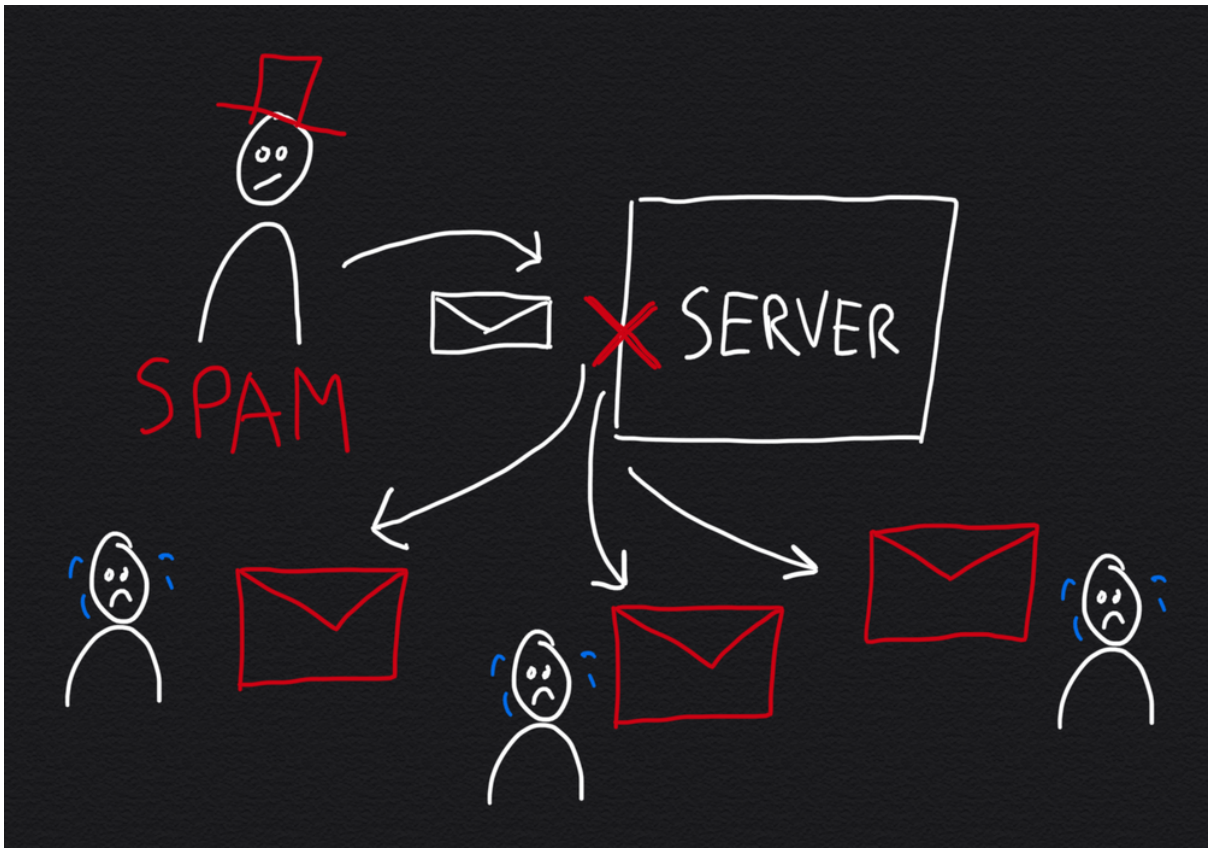
Fighting backscatter spam at server level

Configure Postfix to block spam before it enters the server

Willem L. Middelkoop
Sep. 10, 2019



This month I had to deal with backscatter spam, affecting one of the mail servers I manage. As server engineer I make sure that servers don't send spam and that incoming email gets filtered. Despite all good efforts, this server kept being blacklisted for sending spam to iCloud, Office 365 and Google Gmail for Business (G Suite). Read along to find out how what caused this and how to fix this.

The problem

The server is a Postfix mailserver on Debian GNU/Linux running all the latest updates and security patches. The server is used by legitimate customers that can only use it with proper authentication.

Prevent begin an open relay

Individual user authentication on a mail server is an important measure against abusing the mail server. If spam gets sent, you can see what user is causing the problem by looking at the mail server logs. In technical terms this means you prevent the server to become 'an open relay'. Postfix offers many settings to limit SMTP relay using access control.

Rate limiting (hacked) mail accounts

Server abuse by individual users happens when their computer gets a virus or when their password gets sniffed. That can happen when you access your mail over an insecure WiFi without encryption (TLS/SSL). For situations like this, it's a good idea to rate limit the amount of emails that individual users are allowed to send. You can use individual Postfix configuration parameters (such as *smtpd_client_message_rate_limit, smtpd_client_connection_rate_limit* and *smtpd_client_recipient_rate_limit*) or use a Postfix firewall like Postwfd.

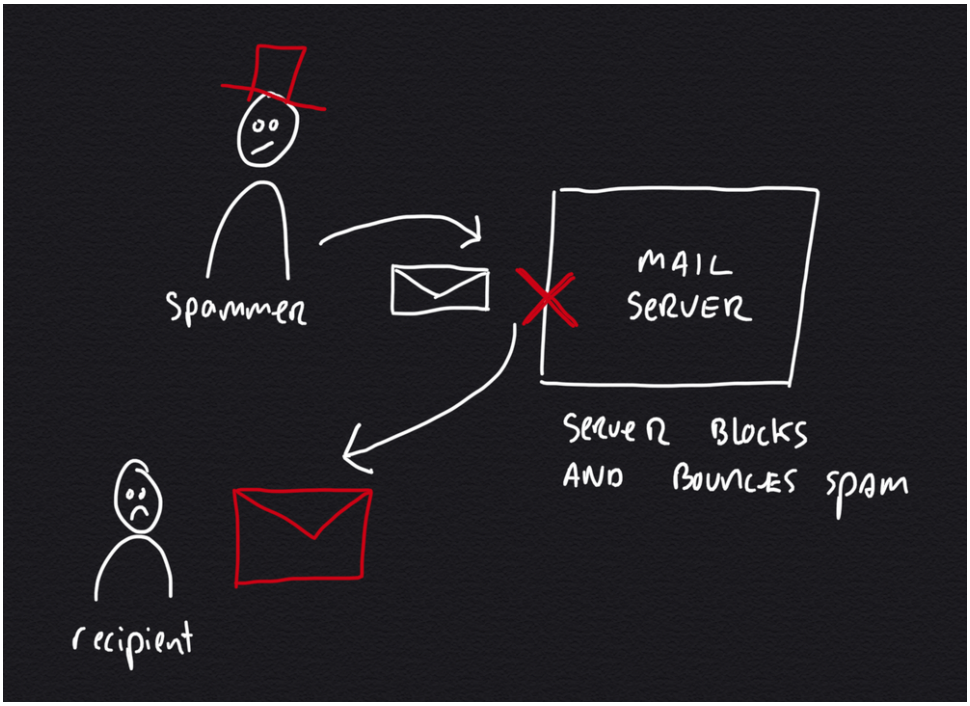
The server gets listed on 'backscatterer.org' black list

Despite all the measures against spam abuse, the server still got listed on `http://backscatter.org` for sending spam. If a server gets listed on a black list, other servers will no longer trust it and consider all messages to be suspicious. This can cause your (legitimate) messages to be seen by other servers as spam (false positives). To understand why this happened, you first have to understand what backscatter spam is.

What is Backscatter spam?

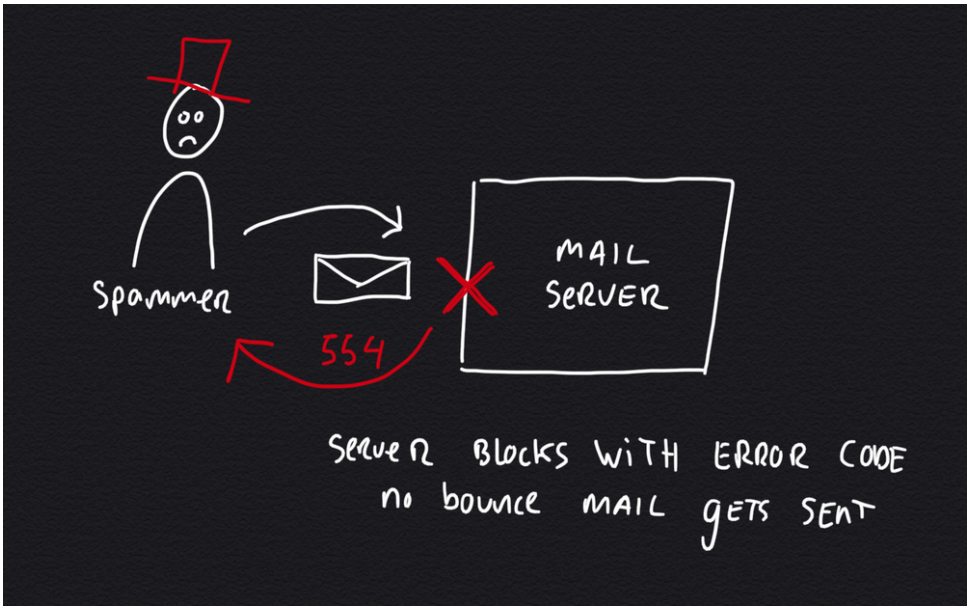
Backscatter spam is incorrectly automated bounce messages sent by the mail server, usually as a side effect of incoming spam.

Backscatter occurs when spammers forge the sender address. They send a mail message that they expect to be bounced using somebody else's address as sender. When the mail message bounces, the mail server returns the message to the falsified sender... causing the server to effectively deliver spam to the sender address.



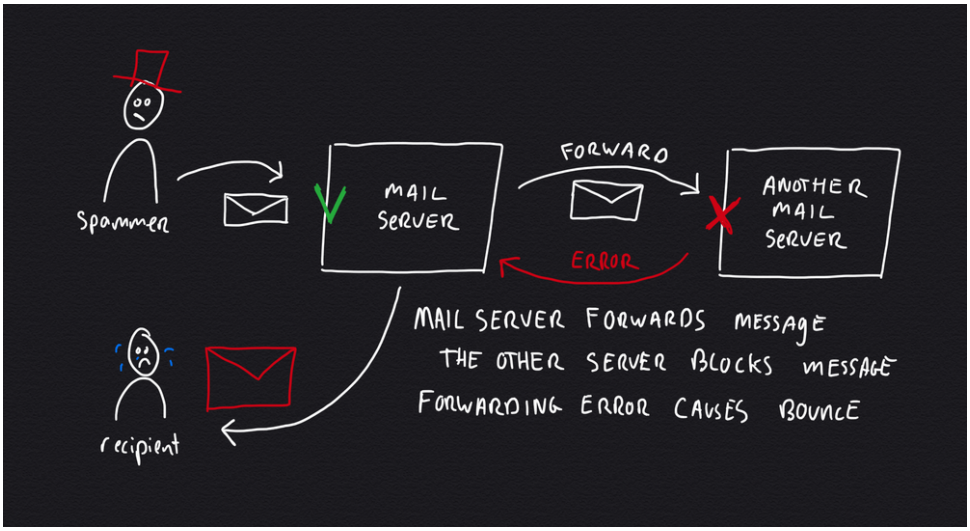
Simple backscatter scenario, mail server bounces message to a falsified sender address

The simple backscatter scenario is where a message gets bounced to the wrong sender. It's a scenario that's simple to solve by configuring the server not to send bounce messages, but to respond with an SMTP error code instead.



Prevent backscatter bounce messages by responding with SMTP error codes instead

Unfortunately there is another (harder) scenario causing your mail server to send backscatter spam. This happens when somebody on your server configures his or her mail account to forward email to another address. The second server might have a more stringent spam filter that causes messages to be blocked upon forward. Your mail server will then inform the original sender that forwarding failed, causing backscatter spam.

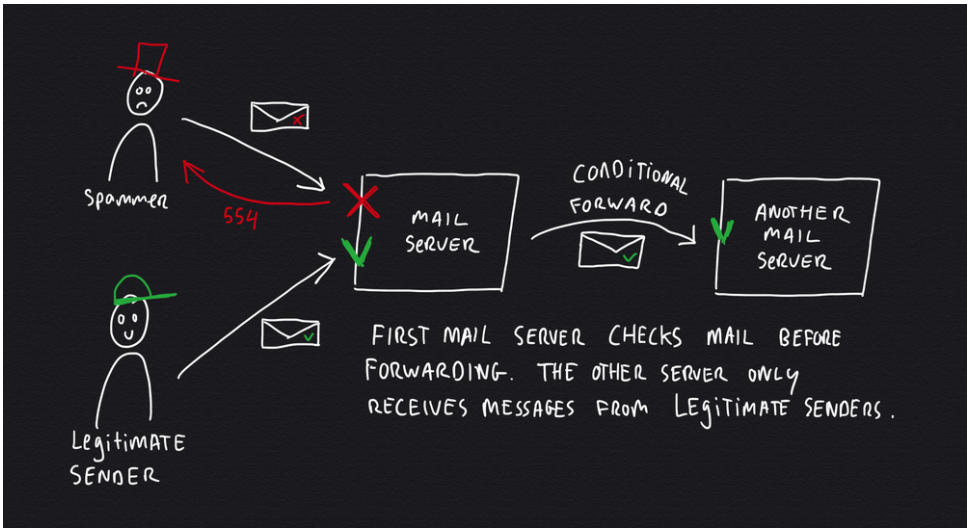


Causing backscatter spam when the first mail server forwards email to another server that blocks message

The forwarding scenario is much harder to solve as you possibly don't control the other mail server or its spam filter settings. What's worse is that there is often a some delay between accepting the message and forwarding it, making it impossible to wait for a response of the other mail server.

Prevent backscatter spam

The only way you can fully prevent backscatter spam is to be very stringent at the first mail server, at the initial SMTP connection stage. This solves the challenging 'forwarding scenario' by being very selective what messages get forwarded.



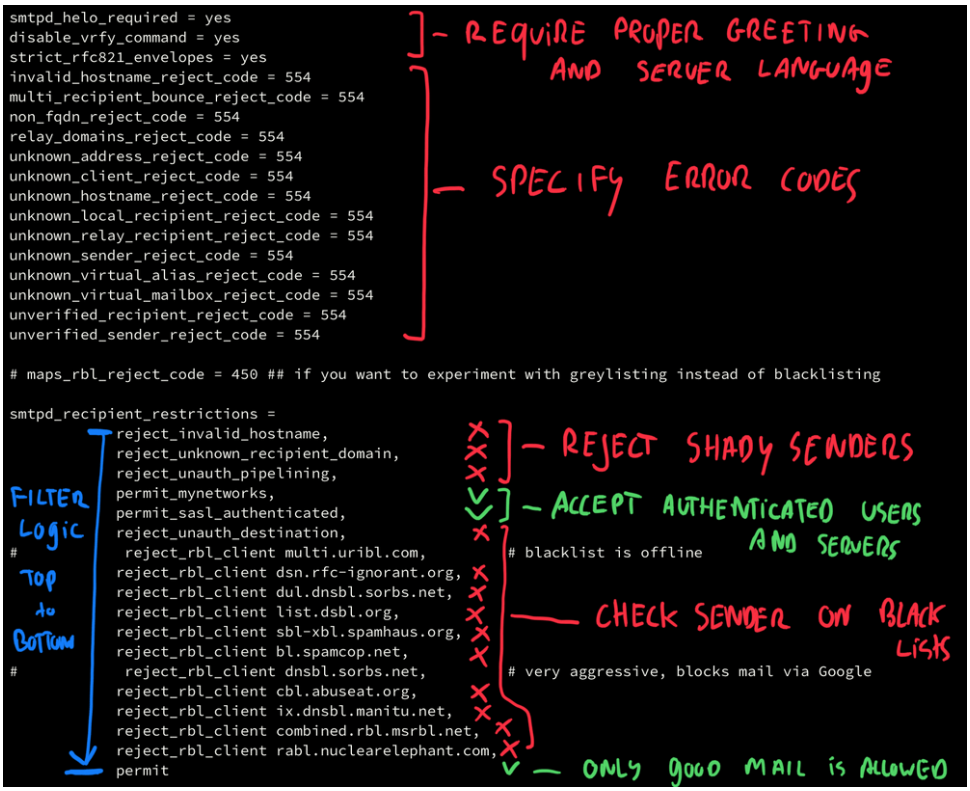
Prevent backscatter spam by checking all messages before forwarding them to another server

How to block spam before it enters the server

The best way to prevent backscatter spam is to block it before it enters the (chain of) server(s).

Traditionally, servers use advanced spam filter such as SpamAssasin to analyse messages one by one. This causes additional load on the mail server.

It's much easier to look at the sender (e.g. if he wears a red hat) for initial filtering. Is the sender on a black list? Does he use a 'normal' mail app, or are messages being send by a virus or malware? You can determine many things by looking at the 'language' the sender speaks.



Implementing stringent SMTP restrictions in Postfix - annotated screenshot from main.cf

In Postfix you can specify SMTP restrictions in the *main.cf* configuration file. There are checks that can be used to block messages. It's a good idea to work from 'easy to hard' in filter logic, starting with things that can be checked without consulting external servers. This reduces network traffic and load. Only if a message passes all checks, it gets permitted for delivery (or forwarding).

Upon delivery you can implement additional complex (and user specific) spam filtering using tools like SpamAssasin or Bayes filtering. By reducing the influx of obvious spam messages, these resource intensive filters can do their work much more efficiently.

Conclusion

Fighting spam at server level is a challenging game as spammers get more creative in abusing mechanisms of mail servers. If your server gets blacklisted for backscatter spam, you need to be wary of forwarding scenarios. Keep an eye on blacklists to see if your server is causing trouble.

Block messages with error codes instead of returning them to the sender (bouncing). Configure the server with a stringent delivery policy. The best way to prevent backscatter spam is to block messages before they enter the mail server, now you know how.