Data Query Service Manual

1.0 - Datafeed Query Service

Your Datafeed Query Account Key is: *26 digit code personal to you*

To query the datafeed query service, your key is added to the DNSBL zone names that you will use in the same way as you would use any normal DNSBLs in your mail server.

If, for example, you have been using `zen.spamhaus.org`, you simply delete `zen.spamhaus.org` and replace it with: `Account Key.zen.dq.spamhaus.net`

2.0 - Testing DNSBL resolution

Your DNS resolver should be able to resolve the DNSBL queries without any particular configuration, as these queries are normal DNS traffic.

Please note that IP lookups (zen, sbl, xbl, pbl) and domain lookups (dbl, zrd) have independent authorization, depending on your contract, you can have both kinds of lookups working of them or only one of them. Perform only the tests that apply to your contract.

We recommend that you perform these tests from your mail servers, which will be the main users'. If this test is not working as expected and your network has a firewall doing inspection of DNS packets, please try disabling that feature to check if it was affecting DNSBL lookups.

Caution:
The datafeed query service uses the domain: spamhaus.NET not spamhaus.ORG.

Do not confuse them as they are not interchangeable.

Do not proceed to the next section until these tests work as expected.

IP lookups

If you purchased the IP lookup service, please use one of the common lookup tools (`nslookup`, `host` or `dig` depending on the operating system and environment) to verify that a A query for

```
2.0.0.127.Account Key.zen.dq.spamhaus.net
```

returns

```
2.0.0.127.Account Key.zen.dq.spamhaus.net. 60 IN A 127.0.0.2
2.0.0.127.Account Key.zen.dq.spamhaus.net. 60 IN A 127.0.0.4
2.0.0.127.Account Key.zen.dq.spamhaus.net. 60 IN A 127.0.0.10
```

Note:
Throughout the worked examples in this manual *`Account Key`* refers to the 26 digit code sent to you.
For you own implementation, please make sure you use the exact key as provided to you.

Domain lookups

If you purchased the domain lookup service, use the same procedure to verify that an A query for

```
dbltest.com.Account Key.dbl.dq.spamhaus.net
```

returns

```
dbltest.com.Account Key.dbl.dq.spamhaus.net. 60 IN A 127.0.1.2
```

and that an A query for

```
zrdtest.com.Account Key.zrd.dq.spamhaus.net
```

returns

```
zrdtest.com.Account Key.zrd.dq.spamhaus.net. 60 IN A 127.0.2.2
```

3.0 - Configuring your mail servers

The last step to do is to configure your mail servers so that they send DNS queries to check if the client IPs are listed on the Spamhaus blocking lists, and reject all mails that satisfy this condition.

For IP lookups, this can be done using the generic BL lookup facility that you find in virtually all the mail server or anti-spam appliance products in use today. Please consult your mail server or appliance documentation for details. Unfortunately, similar lookup facilities for domains are not yet available on all mail server products.

In this section we start with some general indications, valid for all mail servers, and describe more in detail what to do for some common servers and appliances.

3.1 - General indications

In general, this is what you have to do:

1. Make sure that the server/appliance resolves DNS names using your DNS resolver(s), configured as described in section 2.

2. If you purchased the IP lookup service, configure your mail server or anti-spam appliance to use the external blocking list for

IP addresses

```
Account Key.zen.dq.spamhaus.net
```

or

```
Account Key.sbl.dq.spamhaus.net
Account Key.pbl.dq.spamhaus.net
Account Key.xbl.dq.spamhaus.net
```

if you have chosen to go with separate zones; turn every reference to

```
  zone.spamhaus.org
```

into a reference to

```
  Account Key.zone.dq.spamhaus.net.
```

3. If you purchased the domain lookup service, configure your mail server or anti-spam appliance to use the external blocking lists for domains

```
Account Key.dbl.dq.spamhaus.net
```

and

```
Account Key.zrd.dq.spamhaus.net
```

Again, if the configuration includes references to `dbl.spamhaus.org`, replace them with corresponding references to `Account Key.dbl.dq.spamhaus.net`

Below are instructions on how to accomplish this for some specific products.

Note that, in general, DNSBL queries are made at two different stages:
1. During the initial SMTP negotiation, to decide whether the message should be accepted or not (this is before the message headers and body are transmitted);

2. If the message is accepted, its contents (including headers) may be inspected, and IP addresses and domains found there could be looked up using DNSBLs.

In several cases, the second stage is carried on by special software - such as, for instance, SpamAssassin - hooked to the mail server but separate from it, with its own configuration file. Again, all references to zone.spamhaus.org in these configurations must be replaced with corresponding references to *Account Key*.zone.dq.spamhaus.net.

Content analysis should be done with:
Account Key.sbl-xbl.dq.spamhaus.net
or
Account Key.sbl.dq.spamhaus.net.

The former blocks more spam but may generate occasional false positives.

Note: Do not use *Account Key*.zen.dq.spamhaus.net for content analyzers, URL verification, etc. It contains the PBL database which is not designed for this purpose.

3.1.1 - Configuring sendmail

Locate the sendmail.mc file and edit it. Insert the following line (as a single line with no linefeeds):

```
FEATURE(dnsbl, `Account Key.zen.dq.spamhaus.net', `"550 Mail from “ $&{client_addr} "
rejected using zen.spamhaus.org. Please see http://www.spamhaus.org/query/bl?
ip="$&{client_addr}')dnl
```

The location of the line of the file is not important. Produce a *sendmail* configuration file using the *m4* command:

```
m4 sendmail.mc > sendmail.cf
```

then place sendmail.cf into the appropriate directory (usually /etc or /etc/mail) and restart *sendmail*.

Using domain lists such as DBL and ZRD with sendmail requires installing additional software, such as SnertSoft's milter-link or 510 Software Group's milter.

3.1.2 - Configuring postfix

Note: We can not support Postfix releases before 2.8. The following documentation cannot be applied to such old releases. **Note**: The following instructions apply for users with access to both IP and domain lookups. Users with only IP access must omit references to dbl and zrd. Users with only domain access must omit references to zen, sbl, pbl and xbl.

Edit main.cf (usually located in /etc/postfix), and add:
in the list of smtpd_recipient_restrictions.

```
reject_rhsbl_sender            Account key.dbl.dq.spamhaus.net=127.0.1.[2..99],
reject_rhsbl_helo              Account key.dq.spamhaus.net=127.0.1.[2..99],
reject_rhsbl_reverse_client    Account key.dbl.dq.spamhaus.net=127.0.1.[2..99],
reject_rhsbl_sender            Account key.zrd.dq.spamhaus.net=127.0.2.[2..24],
reject_rhsbl_helo              Account key.zrd.dq.spamhaus.net=127.0.2.[2..24]
reject_rhsbl_reverse_client    Account key.zrd.dq.spamhaus.net=127.0.2.[2..24]
reject_rbl_client              Account key.zen.dq.spamhaus.net=127.0.0.[2..255]
```

Then create a file named for instance dnsbl-reply-map containing the lines

```
Account key.sbl.dq.spamhaus.net=127.0.0.[2..255]     $rbl_code Service unavailable; $rbl_class [$rbl_what] blocked
Account key.xbl.dq.spamhaus.net=127.0.0.[2..255]     $rbl_code Service unavailable; $rbl_class [$rbl_what] blocked
Account key.pbl.dq.spamhaus.net=127.0.0.[2..255]     $rbl_code Service unavailable; $rbl_class [$rbl_what] blocked
Account key.sbl-xbl.dq.spamhaus.net=127.0.0.[2..255] $rbl_code Service unavailable; $rbl_class [$rbl_what] blocked
Account key.zen.dq.spamhaus.net=127.0.0.[2..255]     $rbl_code Service unavailable; $rbl_class [$rbl_what] blocked
Account key.dbl.dq.spamhaus.net=127.0.1.[2..99]      $rbl_code Service unavailable; $rbl_class [$rbl_what] blocked
Account key.zrd.dq.spamhaus.net=127.0.2.[2..24]      $rbl_code Service unavailable; $rbl_class [$rbl_what] blocked
```

Create a hash map of it with

```
postmap hash:dnsbl-reply-map
```

then insert

```
rbl_reply_maps = hash:dnsbl-reply-map
```

in main.cf.

Reload postfix.

If you also use the optional postscreen feature, you can use

```
postscreen_dnsbl_sites = Account Key.zen.dq.spamhaus.net=127.0.0.[2..255]
postscreen_dnsbl_reply_map = texthash:/etc/postfix/dnsbl_reply
```

with dnsbl_reply containing the lines

```
Account key.sbl.dq.spamhaus.net     sbl.spamhaus.org
Account key.xbl.dq.spamhaus.net     xbl.spamhaus.org
Account key.pbl.dq.spamhaus.net     pbl.spamhaus.org
Account key.zen.dq.spamhaus.net     zen.spamhaus.org
Account key.dbl.dq.spamhaus.net     dbl.spamhaus.org
Account key.zrd.dq.spamhaus.net     zrd.spamhaus.org
```

Note: ZRD (Zero Reputation Domains) is a domain list containing domains that have been observed for the first time by Spamhaus probes less than 24 hours ago. The possible return codes are the following:

```
 127.0.2.2 : domain first observed between 0 and 2 hours ago
 127.0.2.3 : domain first observed between 2 and 3 hours ago
 127.0.2.4 : domain first observed between 3 and 4 hours ago
 127.0.2.5 : domain first observed between 4 and 5 hours ago
 127.0.2.6 : domain first observed between 5 and 6 hours ago
 127.0.2.7 : domain first observed between 6 and 7 hours ago
 127.0.2.8 : domain first observed between 7 and 8 hours ago
 127.0.2.9 : domain first observed between 8 and 9 hours ago
127.0.2.10 : domain first observed between 9 and 10 hours ago
127.0.2.11 : domain first observed between 10 and 11 hours ago
127.0.2.12 : domain first observed between 11 and 12 hours ago
127.0.2.13 : domain first observed between 12 and 13 hours ago
127.0.2.14 : domain first observed between 13 and 14 hours ago
127.0.2.15 : domain first observed between 14 and 15 hours ago
127.0.2.16 : domain first observed between 15 and 16 hours ago
127.0.2.17 : domain first observed between 16 and 17 hours ago
127.0.2.18 : domain first observed between 17 and 18 hours ago
127.0.2.19 : domain first observed between 18 and 19 hours ago
127.0.2.20 : domain first observed between 19 and 20 hours ago
127.0.2.21 : domain first observed between 20 and 21 hours ago
127.0.2.22 : domain first observed between 21 and 22 hours ago
127.0.2.23 : domain first observed between 22 and 23 hours ago
127.0.2.24 : domain first observed between 23 and 24 hours ago
```

After 24 hours from the first observation, domains exit the list. In the configuration above, we assume that all these domains have to be considered. If one wishes to block only the domains first observed up to N hours ago (with N an integer number between 2 and 24), the ’24’ appearing in the example configuration can simply be replaced with the chosen N (in both main.cf and dnsbl-reply-map).

3.1.3 - Configuring qmail

First of all, you must have ucspi-tcp installed to enable you to run *rblsmtpd*, the qmail module that does DNS lookups to blocking lists and uses the results to allow or disallow SMTP connections.

Edit the `run` file (usually located in `/var/qmail/supervise/qmail-smtpd`) and locate the line invoking *tcpserver* at the bottom of the file. It will be something like

```
/usr/local/bin/tcpserver (...options...) \
0 smtp \
qmail-smtpd
```

(the details can differ in your installation, and you may see `25` in place of `smtp`). Insert the invocation of `rblsmtpd` just before the `qmail-smtpd` argument as follows:

```
/usr/local/bin/tcpserver (...options...) \
0 smtp \
/usr/local/bin/rblsmtpd -b -r Account Key.zen.dq.spamhaus.net \
qmail-smtpd
```

With this setting, an incoming SMTP connection will launch *rblsmtpd*, which will check if the client IP is listed in our databases. If it is listed *rblsmtpd* handles the rejection dialogue, while if it is not listed *rblsmtpd* launches `qmail-smtpd` for normal mail delivery.

Note: Some qmail users have reported that the BL checks are made also for outgoing mail submitted by legitimate users of the mail server using authenticated SMTP. In these cases, the client IP is not a mail server and is likely to be listed in the PBL database.

Therefore, users would not be able to send mail. The standard way to solve this problem is to run a separate qmail instance that listens on port 587, accepts only authenticated mail submissions and does not use *rblsmtpd*.

3.1.4 - Configuring SpamAssassin

In this section we describe how to configure SpamAssassin so that it sends a query to your local server rather than to the public servers of The Spamhaus Project.

It is assumed that your Datafeed Query Service subscription includes both IP data and Domain data.

These instructions assume that your SpamAssassin version is at least 3.4.1, released in April 2015. If you are running an earlier release, please upgrade.

Warning: SpamAssassin 3.4.1 has an important bug that needs to be patched if the release of the Net::DNS Perl package installed on your system is 1.01 or larger.
If you installed SpamAssassin as an O/S package, the bug may have been fixed already.

If you installed it from sources, the bug is present. In all cases, locate the directory where the `DnsResolver.pm` file is located and run the following commands:

```
if ! grep -q '$packet->header->rd(1)' DnsResolver.pm then # apply patch
cp -p DnsResolver.pm DnsResolver.pm.orig patch << 'EOF'

--- DnsResolver.pm 2015-04-28 19:56:49.000000000 +0000
+++ DnsResolver.pm.new 2015-07-20 18:24:48.000000000 +0000 @@ -592,6 +592,9 @@
};
if ($packet) {
# RD flag needs to be set explicitly since Net::DNS 1.01, Bug 7223 $packet->header-
>rd(1);
# my $udp_payload_size = $self->{res}->udppacketsize;
my $udp_payload_size = $self->{conf}->{dns_options}->{edns}; if ($udp_payload_size &&
$udp_payload_size > 512) {
+ + +
EOF else
echo "DnsResolver.pm has already been patched." fi
```

It will not be necessary to do this on the forthcoming SpamAssassin 3.4.2.

Now go to the SpamAssassin configuration directory, which is usually `/etc/mail/spamassassin` or `/etc/spamassassin`. Insert the following definitions (overriding SpamAssassin's own definitions, referring to the Spamhaus Project public service) in the file `local.cf`:

```
header __RCVD_IN_ZEN eval:check_rbl('zen','Account Key.zen.dq.spamhaus.net.')
header RCVD_IN_XBL eval:check_rbl(‘zen-lastexternal','Account Key.zen.dq.spamhaus.net.','127.0.0.[45678]')
header RCVD_IN_PBL eval:check_rbl('zen-lastexternal', ‘Account Key.zen.dq.spamhaus.net.', '127.0.0.1[01]')
ifplugin Mail::SpamAssassin::Plugin::URIDNSBL
uridnssub URIBL_SBL                  Account key.zen.dq.spamhaus.net. A 127.0.0.2
uridnsbl URIBL_SBL_A                 Account Key.sbl.dq.spamhaus.net. A
urirhssub URIBL_DBL_SPAM             Account Key.dbl.dq.spamhaus.net. A 127.0.1.2
urirhssub URIBL_DBL_PHISH            Account Key.dbl.dq.spamhaus.net. A 127.0.1.4
urirhssub URIBL_DBL_MALWARE          Account Key.dbl.dq.spamhaus.net. A 127.0.1.5
urirhssub URIBL_DBL_BOTNETCC         Account Key.dbl.dq.spamhaus.net. A 127.0.1.6
urirhssub URIBL_DBL_ABUSE_SPAM       Account Key.dbl.dq.spamhaus.net. A 127.0.1.102
urirhssub URIBL_DBL_ABUSE_REDIR      Account Key.dbl.dq.spamhaus.net. A 127.0.1.103
urirhssub URIBL_DBL_ABUSE_PHISH      Account Key.dbl.dq.spamhaus.net. A 127.0.1.104
urirhssub URIBL_DBL_ABUSE_MALW       Account Key.dbl.dq.spamhaus.net. A 127.0.1.105
urirhssub URIBL_DBL_ABUSE_BOTCC      Account Key.dbl.dq.spamhaus.net. A 127.0.1.106
urirhssub URIBL_DBL_ERROR            Account Key.dbl.dq.spamhaus.net. A 127.0.1.255
if can(Mail::SpamAssassin::Plugin::URIDNSBL::has_tflags_domains_only)
urihsbl   URIBL_ZRD                  Account key.zrd.dq.spamhaus.net A
Body.     URIBL_ZRD                  eval:checkuridnsbl( URIBL_ZRD)
Describe  URIBL_ZRD                  Contains a URL listed in the Spamhaus ZRD blocklist
tflags.   URIBL_ZRD                  net domains_only
score     URIBL_ZRD                  2.5
endif # if can
endif   # Mail::SpamAssasin::Plugin::URIDNSBL
```

Restart SpamAssassin.

For further information we recommend that you consult the SpamAssassin documentation.

3.2 - Testing the mail server

Once you are convinced that your mail server setup is complete, you can try the testing service kindly maintained by Crynwr Software. This service consists of a robot that will send you email from a listed IP to verify that your server blocks the message.

The testing service is available independently for SBL, PBL and XBL (SBL tests are initiated by 192.203.178.107, PBL tests by 192.203.178.178 and XBL tests by 192.203.178.138).

To activate the robot, you have to send an email (content or subject do not matter) to the special addresses `nelson-sbl-test@crynwr.com` (for SBL testing), `nelson-pbl-test@crynwr.com` (for PBL testing), or `nelson-xbl- test@crynwr.com` (for XBL testing).

This email must be sent from the IP address used by your mail server to receive mail, because the robot can test exclusively the IP address that originated the testing request. If you have difficulties in doing that - - for instance because your outbound mail goes out from a different IP, or because you are using multiple IP aliases -- you will have to send mail using telnet ‘by hand’, making sure that the source IP address of your connection coincides with the IP address where your mail server is listening for connections.

Also make sure that you are using a functional email address under your control as sender, as the results of the tests will be mailed to this address.

If the test is successful, you will get back from the robot an email containing the SMTP dialogue between the robot and your server. If the test is unsuccessful, you will get two mails: The mail containing the SMTP dialogue, and the test mail that the robot managed to send you in spite of being listed on the BL you are testing.

If all works, congratulations: Your setup work has now finished. Enjoy the missing spam!