

SMTP Service Extension  
for Remote Message Queue Starting  
<[draft-dewinter-queue-start-02.txt](#)>

June 12, 1996

## **1. Status of this Memo**

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

To learn the current status of any Internet-Draft, please check the "ltd-abstracts.txt" listing contained in the Internet-Drafts Shadow Directories on ds.internic.net (US East Coast), nic.nordu.net (Europe), ftp.isi.edu (US West Coast), or munnari.oz.au (Pacific Rim).

## **2. Abstract**

This memo defines an extension to the SMTP service whereby an SMTP client and server may interact to give the server an opportunity to start the processing of its queues for messages to go to a given host. This is an attempt to fix the security problems with the TURN by creating a new command ETRN that places the emphasis on the server instead of the connection. This extension is meant to be used in startup conditions as well as for mail nodes that have transient connections to their service providers.

## **3. Introduction**

The TURN command was a valid attempt to address the problem of having to start the processing for the mail queue on a remote machine. However, the TURN command presents a large security loophole. As there is no verification of the remote host name, the TURN command could be used by a rogue system to download the mail for a site other than itself.

Therefore, this memo introduces the ETRN command. This command

uses the mechanism defined in [4] to define extensions to the SMTP service whereby a client ("sender-SMTP") may request that the server ("receiver-SMTP") start the processing of its mail queues for messages that are waiting at the server for the client machine. If any messages are at the server for the client, then the server should create a new SMTP session and send the messages at that time.

#### **4. Framework for the ETRN Extension**

The following service extension is therefore defined:

- (1) the name of the SMTP service extension is "Remote Queue Processing Declaration";
- (2) the EHLO keyword value associated with this extension is "ETRN", with no associated parameters;
- (3) one additional verb, ETRN, with a single parameter that specifies the name of the client(s) to start processing for;
- (4) no additional SMTP verbs are defined by this extension.

The remainder of this memo specifies how support for the extension affects the behavior of an SMTP client and server.

#### **5. The Remote Queue Processing Declaration service extension**

To save money, many small companies want to only maintain transient connections to their service providers. In addition, there are some situations where the client sites depend on their mail arriving quickly, so forcing the queues on the server belonging to their service provider may be more desirable than waiting for the retry timeout to occur.

Both of these situations could currently be fixed using the TURN command defined in [1], if it were not for a large security loophole in the TURN command. As it stands, the TURN command will reverse the direction of the SMTP connection and assume that the remote host is being honest about what its name is. The security loophole is that there is no documented stipulation for checking the authenticity of the remote host name, as given in the HELO or EHLO command. As such, most SMTP and ESMTP implementations do not implement the TURN command to avoid this security loophole.

This has been addressed in the design of the ETRN command. This extended turn command was written with the points in the first paragraph in mind, yet paying attention to the problems that currently exist with the TURN command. The security loophole is avoided by asking the server to start a new connection aimed at the specified client.

In this manner, the server has a lot more certainty that it is talking to the correct SMTP client. This mechanism can just be seen as a more immediate version of the retry queues that appear in most SMTP implementations. In addition, as this command will take a single parameter, the name of the remote host(s) to start the queues for, the server can decide whether it wishes to respect the request or deny it for any local administrative reasons.

## **6. Definitions**

Remote queue processing means that using an SMTP or ESMTP connection, the client may request that the server start to process parts of its messaging queue. This processing is performed using the existing SMTP infrastructure and will occur at some point after the processing is initiated.

The server host is the node that is responding to the ETRN command.

The client host is the node that is initiating the ETRN command.

The remote host name is defined to be a plain-text field that specifies a name for the remote host(s). This remote host name may also include an alias for the specified remote host or special commands to identify other types of queues.

## **7. The extended ETRN command**

The extended ETRN command is issued by the client host when it wishes to start the SMTP queue processing of a given server host. The syntax of this command is as follows:

```
ETRN [<option character>]<node name><CR><LF>
```

This command may be issued at any time once a session is established, as long as there is not a transaction occurring. Thus, this command is illegal between a MAIL FROM: command and the end of the DATA commands and responses.

The specified node name must be a fully qualified domain name for the node, which may refer to a CNAME or MX pointer in the DNS. If an alias is used for the node, multiple ETRN commands may be needed to start the processing for the node as it may be listed at the remote site under multiple names. This can also be addressed using the options discussed in [section 7.3](#).

The option character under normal circumstances is not used.

### **7.1 Server action on receipt of the extended ETRN command**

When the server host receives the ETRN command, it should have a look at the node name that is specified in the command and make a local decision if it should honour the request. If not, the appropriate error

codes should be returned to the client.

Otherwise, the server host should force its retry queues to start sending messages to that remote site, using another SMTP connection. At the moment, there is no requirement that a connection must occur, or that the connection must occur within a given time frame. This should be noted in the case where there are no messages for the client host at the server host and only the 250 response is used.

Since the processing of the queues may take an indeterminate amount of time, this command should return immediately with a response to the client host. The valid return codes for this command are:

```
250 OK, queuing for node <x> started
251 OK, no messages waiting for node <x>
252 OK, pending messages for node <x> started
253 OK, <n> pending messages for node <x> started
458 Unable to queue messages for node <x>
459 Node <x> not allowed: <reason>
500 Syntax Error
501 Syntax Error in Parameters
```

The 250 response code does not indicate that messages will be sent to the system in question, just that the queue has been started and some action will occur. If the server is capable of supporting it, the 251, 252 or 253 response codes should be used to give more information to the client side. In this case, if there are messages waiting for the client side node, a check can be performed using these responses codes as an indication of when there are no more pending messages in the queue for that node.

The 458 and 459 result codes should be used to give more information back to the client host as to why the action was not performed. If the syntax of the request is not correct, then the 500 and 501 result codes should be used.

## **7.2 Client action on receiving response to extended ETRN command**

If one of the 500 level error codes (550 or 551) are sent, the client should assume that the protocol is not supported in the remote host or that the protocol has not been implemented correctly on either the client or server host. In this case, multiple ETRN commands (dealing with the aliases for the system) should not be sent.

If the 250 response is received, then the client host can assume that the server host found its request to be satisfactory and it will send any queued messages. This process may involve going through a very large retry queue, and may take some time.

If the 400 level response is received, then the client can assume that the server supports the command, but for some local reason does not want to accept the ETRN command as is. In most cases, it will mean that there is a list of nodes that it will accept the command from and the current client is not on that list. The 459 response code is presented to allow for a more in-depth reason as to why the remote queuing cannot be started.

### **7.3 Use Of ETRN to release mail for a subdomain or queue**

If the requesting server wishes to release all of the mail for a given subdomain, a variation on the ETRN command can be used. To perform this request, the option character '@' should be used in front of the node name. In this manner, any domain names that are formed with a suffix of the specified node name are released.

For example, if the command ETRN @foo.com was issued, then any accumulated mail for fred.foo.com, a.b.c.d.e.f.g.foo.com or foo.com may be released. It should be noted that the receiving side of the ETRN command should make a decision based on the client in question and only allow certain combinations for each of the nodes. This is more of a security issue than anything else.

In a similar vein, it might be necessary under some circumstances to release a certain queue, where that queue does not correspond to a given domain name. To this end, the option character '#' can be used to force the processing of a given queue. In this case, the node name would be used as a queue name instead, and its syntactical structure would be dependant on the receiving server. An example of this would be using the command ETRN #uucp to force the flush of a UUCP queue. Note that the use of this option is entirely a local matter and there is no way for a client to find a list of any such queues that exist.

## **8. Minimal usage**

A "minimal" client may use this extension with its host name to start the queues on the server host. This minimal usage will not handle cases where mail for 'x.y' is sent to 's.x.y'.

A minimal server may use this extensions to start the processing of the queues for all remote sites. In this case, the 458 error response will not be seen, and it should always return the 250 response as it will always try and start the processing for any request.

## **9. Example**

The following example illustrates the use of remote queue processing with some permanent and temporary failures.

```

S: <wait for connection on TCP port 25>
C: <open connection to server>
S: 220 sigurd.innosoft.com -- Server SMTP (PMDf V4.2-6 #1992)
C: EHLO ymir.claremont.edu
S: 250-sigurd.innosoft.com
S: 250-EXPN
S: 250-HELP
S: 250 ETRN
C: ETRN
S: 500 Syntax Error
C: ETRN localname
S: 501 Syntax Error in Parameters
C: ETRN uu.net
S: 458 Unable to queue messages for node uu.net
...

C: ETRN sigurd.innosoft.com
S: 250 OK, queuing for node sigurd.innosoft.com started
C: ETRN innosoft.com
S: 250 OK, queuing for node innosoft.com started

OR

C: ETRN sigurd.innosoft.com
S: 251 OK, no messages waiting for node sigurd.innosoft.com
C: ETRN innosoft.com
S: 252 OK, pending messages for node innosoft.com started
C: ETRN mysoft.com
S: 253 OK, 14 pending messages for node mysoft.com started

...
C: ETRN foo.bar
S: 459 Node foo.bar not allowed: Unable to resolve name.
...
C: QUIT
S: 250 Goodbye

```

## **10. Security considerations**

This command does not compromise any security considerations of any existing SMTP or ESMTP protocols as it merely shortens the time that a client needs to wait before their messages are retried.

Precautions should be taken to make sure that any client server can only use the @ and # option characters for systems that make sense. Failure to implement some kind of sanity checking on the parameters could lead to congestion. This would be evident if a person asking to release @com, which would release mail for any address that ended with com.

## **11. Acknowledgements**

This document was created with lots of support from the users of our products, who have given some input to the functionality that they would like to see in the software that they bought.

## **11. References**

- [1] J. B. Postel. Simple Mail Transfer Protocol. Request for Comments 821, August 1982.
- [2] Klensin, J., WG Chair, Freed, N., Editor, Rose, M., Stefferud, E., and D. Crocker, "SMTP Service Extensions" [RFC 1425](#), United Nations University, Innosoft International, Inc., Dover Beach Consulting, Inc., Network Management Associates, Inc., The Branch Office, February 1993.

## **12. Author addresses**

Jack De Winter  
Wildbear Consulting, Inc.

**17 Brock Street**  
Kitchener, Ontario, Canada  
N2M 1X2  
tel: +1 519 576 3873  
email: jack@wildbear.on.ca