

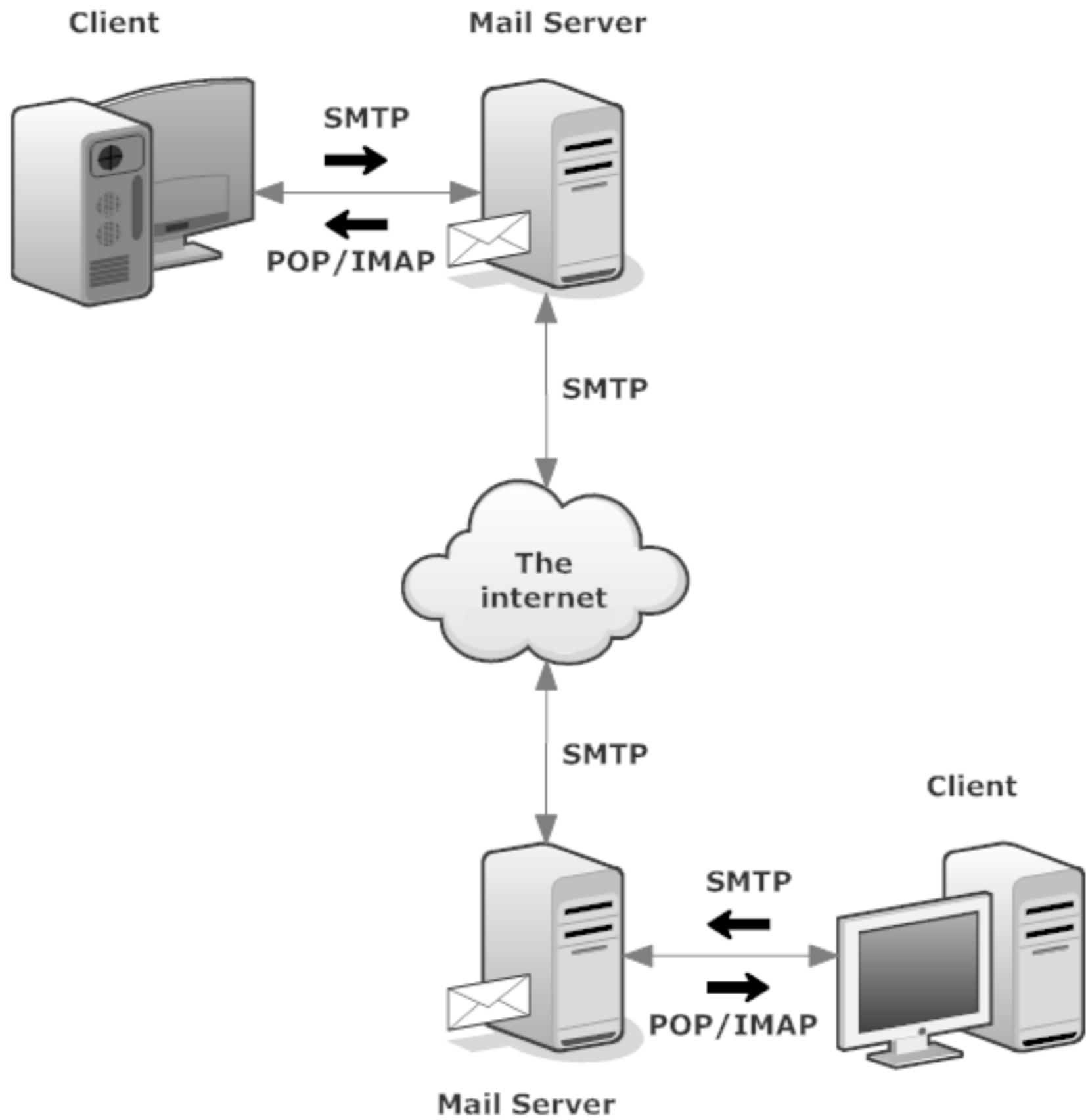
Email

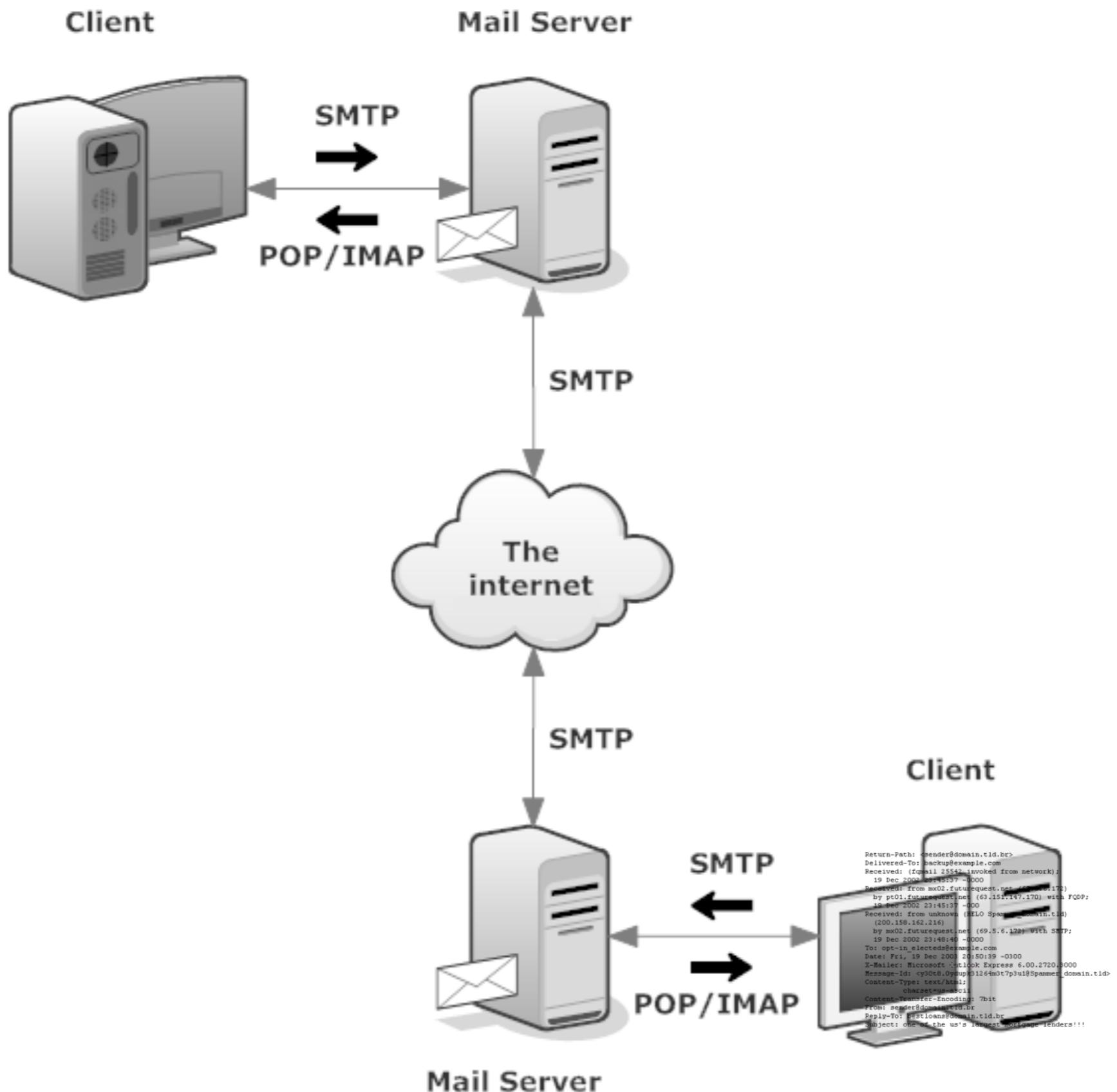
Ew

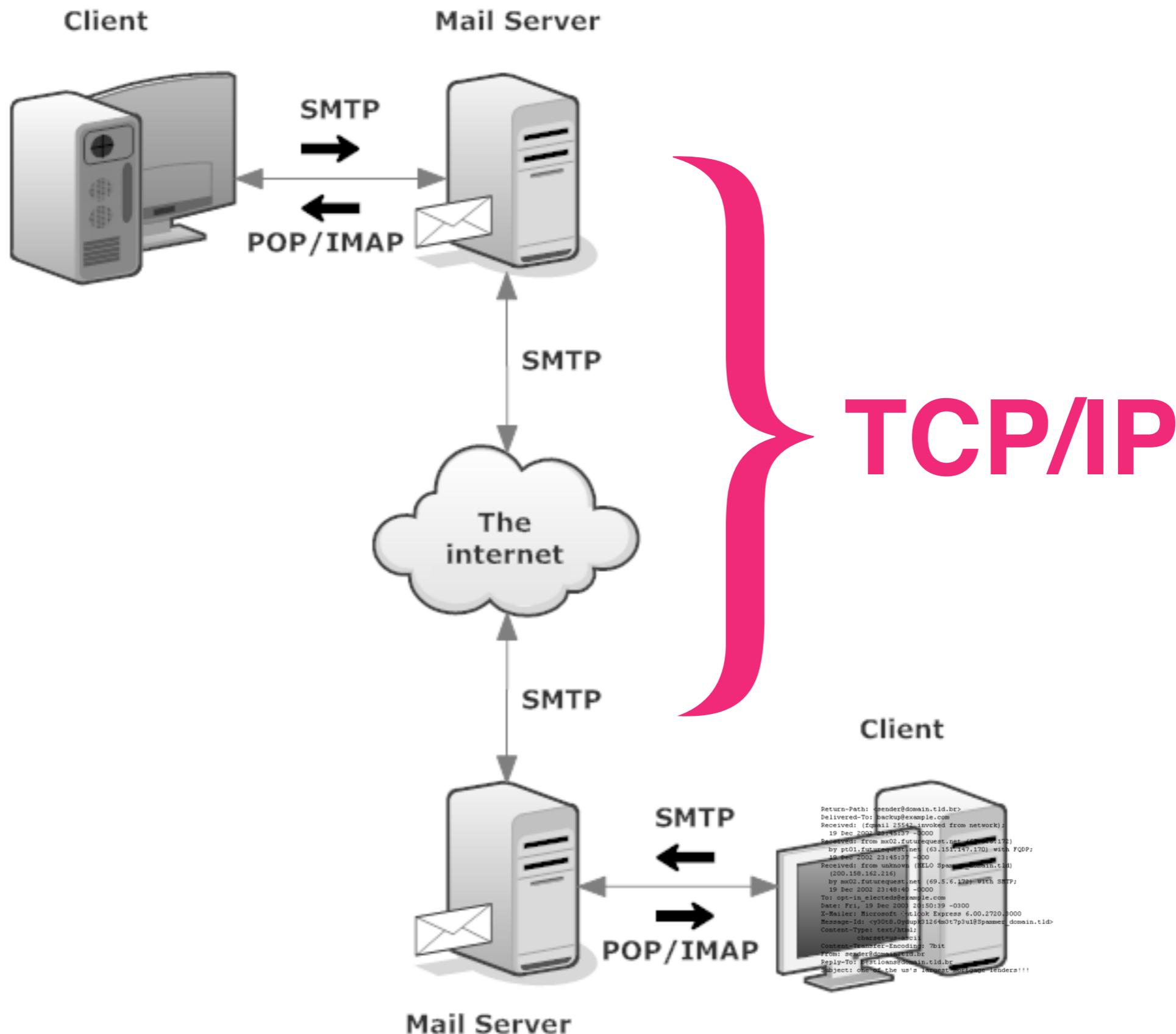


Email+SparkPost+Python

Slightly Less Ew







Return-Path: <sender@domain.tld.br>
Delivered-To: backup@example.com
Received: (fqmail 25542 invoked from network);
19 Dec 2002 23:45:37 -0000
Received: from mx02.futurequest.net (69.5.6.172)
by pt01.futurequest.net (63.151.147.170) with FQDP;
19 Dec 2002 23:45:37 -000
Received: from unknown (HELO Spammer_domain.tld)
(200.158.162.216)
by mx02.futurequest.net (69.5.6.172) with SMTP;
19 Dec 2002 23:48:40 -0000
To: opt-in_electeds@example.com
Date: Fri, 19 Dec 2003 20:50:39 -0300
X-Mailer: Microsoft Outlook Express 6.00.2720.3000
Message-Id: <y30t8.0ydupk31264m3t7p3u1@Spammer_domain.tld>
Content-Type: text/html;
charset=us-ascii
Content-Transfer-Encoding: 7bit
From: sender@domain.tld.br
Reply-To: bestloans@domain.tld.br
Subject: one of the us's largest mortgage lenders!!!

Return-Path: <sender@domain.tld.br>
Delivered-To: backup@example.com
Received: (fqmail 25542 invoked from network);
19 Dec 2002 23:45:37 -0000
Received: from mx02.futurequest.net (69.5.6.172)
by pt01.futurequest.net (63.151.147.170) with FQDP;
19 Dec 2002 23:45:37 -000
Received: from unknown (H [REDACTED]@Spammer_domain.tld)
(200.158.162.216)
by mx02.futurequest.net (69.5.6.172) with SMTP;
19 Dec 2002 23:48:41 -000
To: opt-in_electeds@elect...
Date: Fri, 19 Dec 2003 20:48:41 -0300
X-Mailer: Microsoft Outlook Express 6.00.2720.3000
Message-Id: <y30t8.0ydupk31264m3t7p3u1@Spammer_domain.tld>
Content-Type: text/html;
charset=us-ascii
Content-Transfer-Encoding: 7bit
From: sender@domain.tld.br
Reply-To: bestloans@domain.tld.br
Subject: one of the us's largest mortgage lenders!!!

```
# Import smtplib for the actual sending function
import smtplib

# Import the email modules we'll need
from email.mime.text import MIMEText

# Create a text/plain message
msg = MIMEText("Hello, this is my message.")

msg[ 'Subject' ] = 'Hello there'
msg[ 'From' ] = 'brandon@aubie.ca'
msg[ 'To' ] = 'sally@company.com'

# Send the message via our own SMTP server.
s = smtplib.SMTP('localhost')
s.send_message(msg)
s.quit()
```

Based on: <https://docs.python.org/3.5/library/email-examples.html>

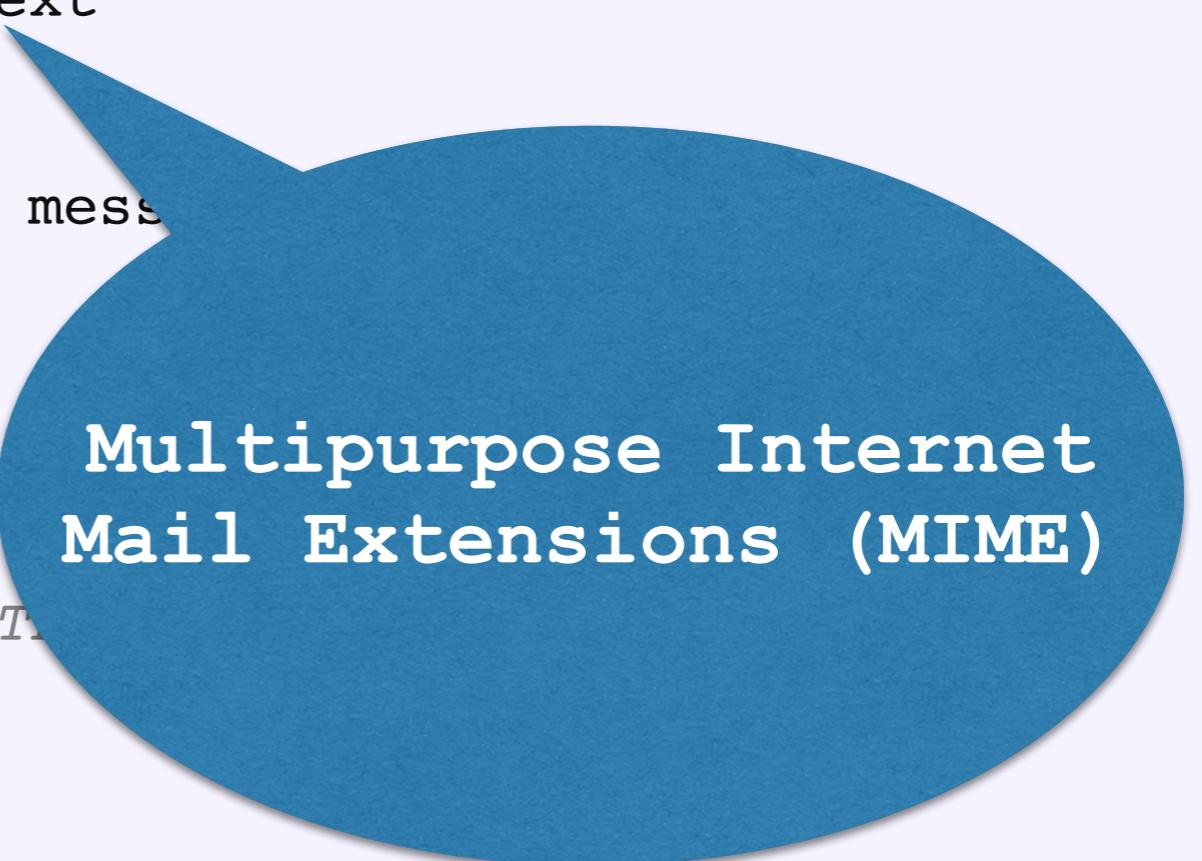
```
# Import smtplib for the actual sending function
import smtplib

# Import the email modules we'll need
from email.mime.text import MIMEText

# Create a text/plain message
msg = MIMEText("Hello, this is my mess

msg[ 'Subject' ] = 'Hello there'
msg[ 'From' ] = 'brandon@aubie.ca'
msg[ 'To' ] = 'sally@company.com'

# Send the message via our own SMTP server.
s = smtplib.SMTP('localhost')
s.send_message(msg)
s.quit()
```

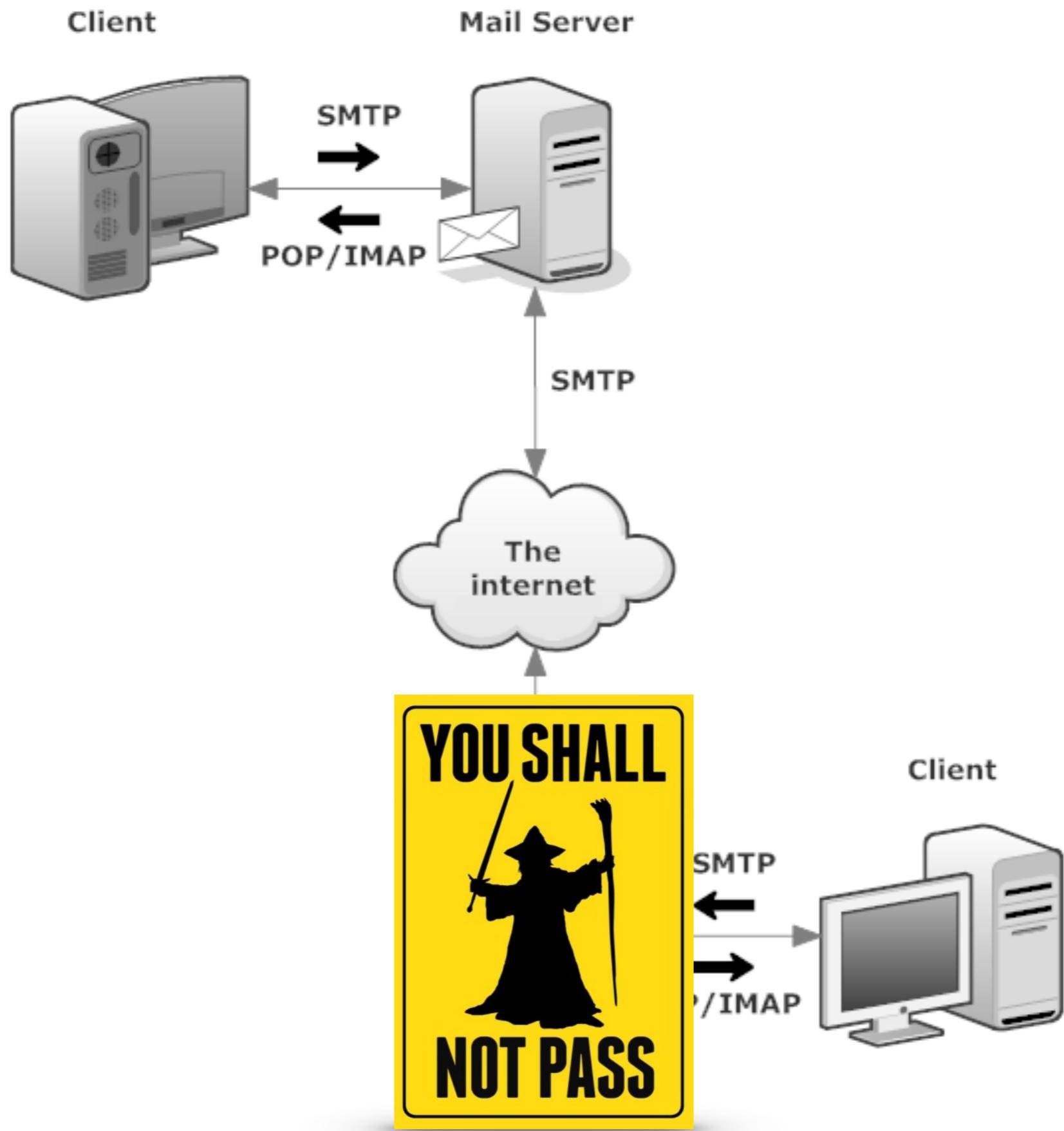


Multipurpose Internet Mail Extensions (MIME)

A scene from Toy Story featuring Woody and Buzz Lightyear. Woody, on the left, has a worried expression and is looking towards the right. Buzz, on the right, is in his signature space ranger pose with arms raised, wearing a green and purple suit with a "HOO-HAAH" patch on his shoulder. The background shows a wooden floor and a wall with a framed picture.

SPAM

SPAM EVERYWHERE



Email

Ew



Search ID: rhan144

Hagen

EVER SINCE THEY INSTALLED THE SPAM FILTER,
I HAVEN'T HAD A SINGLE LETTER.

Q·RESERVE

Sending close to 1000 emails per day
Academic, healthcare and corporate



Reply

Forward

Add note

Start timer



user emails

Susannah-rankin reported 3 months ago (Tue, 22 Mar at 3:21 PM) via Email

To: hello@qreserve.com

Hi,

One of my users is trying to reset her password and she never receives the email to confirm reset. We've tried it a few times, and it does not appear to be blocked by her spamblocker. Help?

Thanks,
Susannah

How to pass a spam filter

- Use correct spelling and grammar
- Don't "look" like SPAM
- Send mail from a trusted SMTP server
- Respect SMTP server requests (busy, full, try later)
- Use the Sender Policy Framework (SPF)
- Use Domain Key Identified Mail (DKIM)



SPF: Sender Policy Framework

Lists which SMTP servers a domain *should* be sending from.

DKIM: Domain Keys Identified Mail

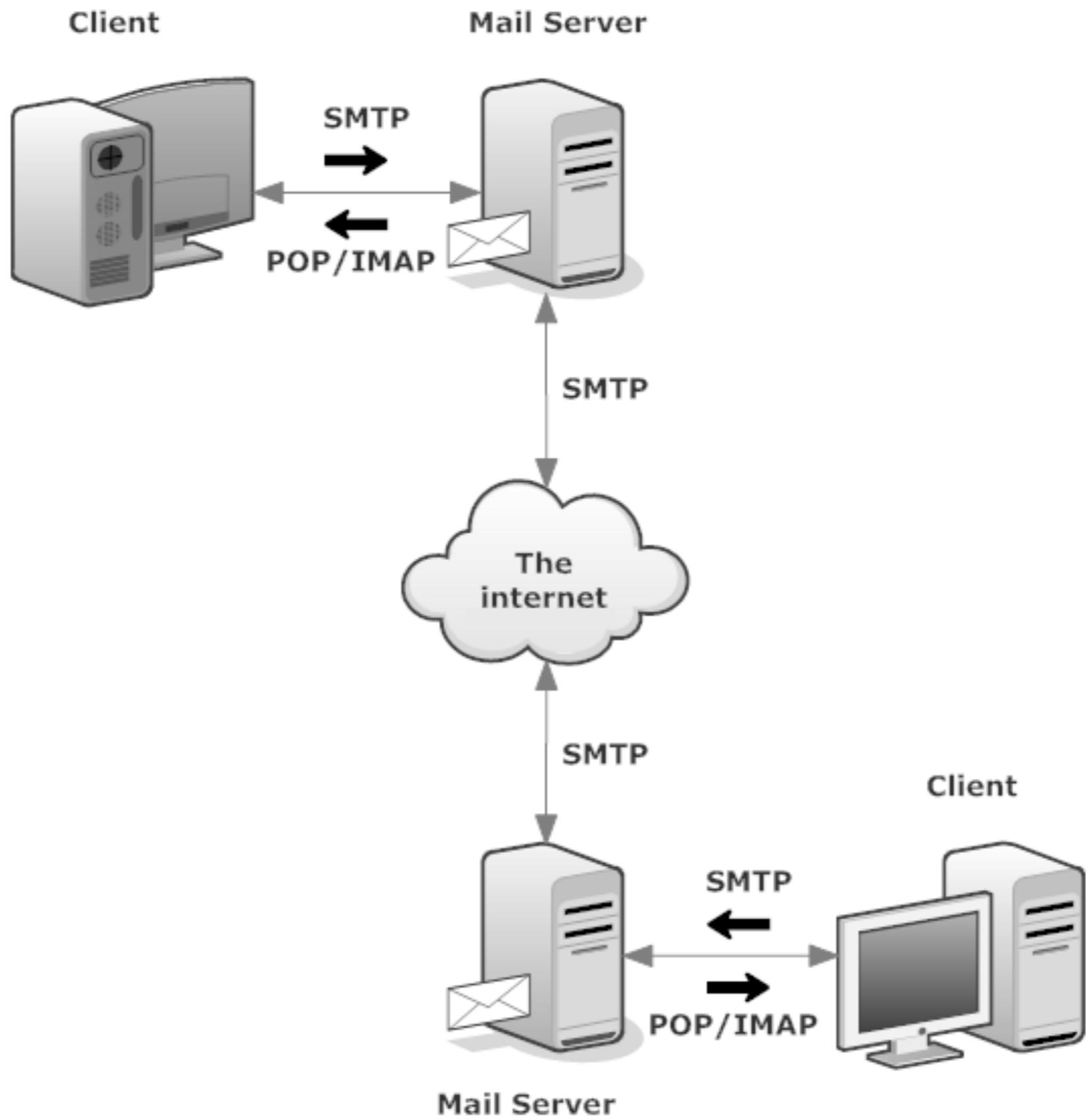
Cryptographically sign emails from the server using public/private key signing techniques.

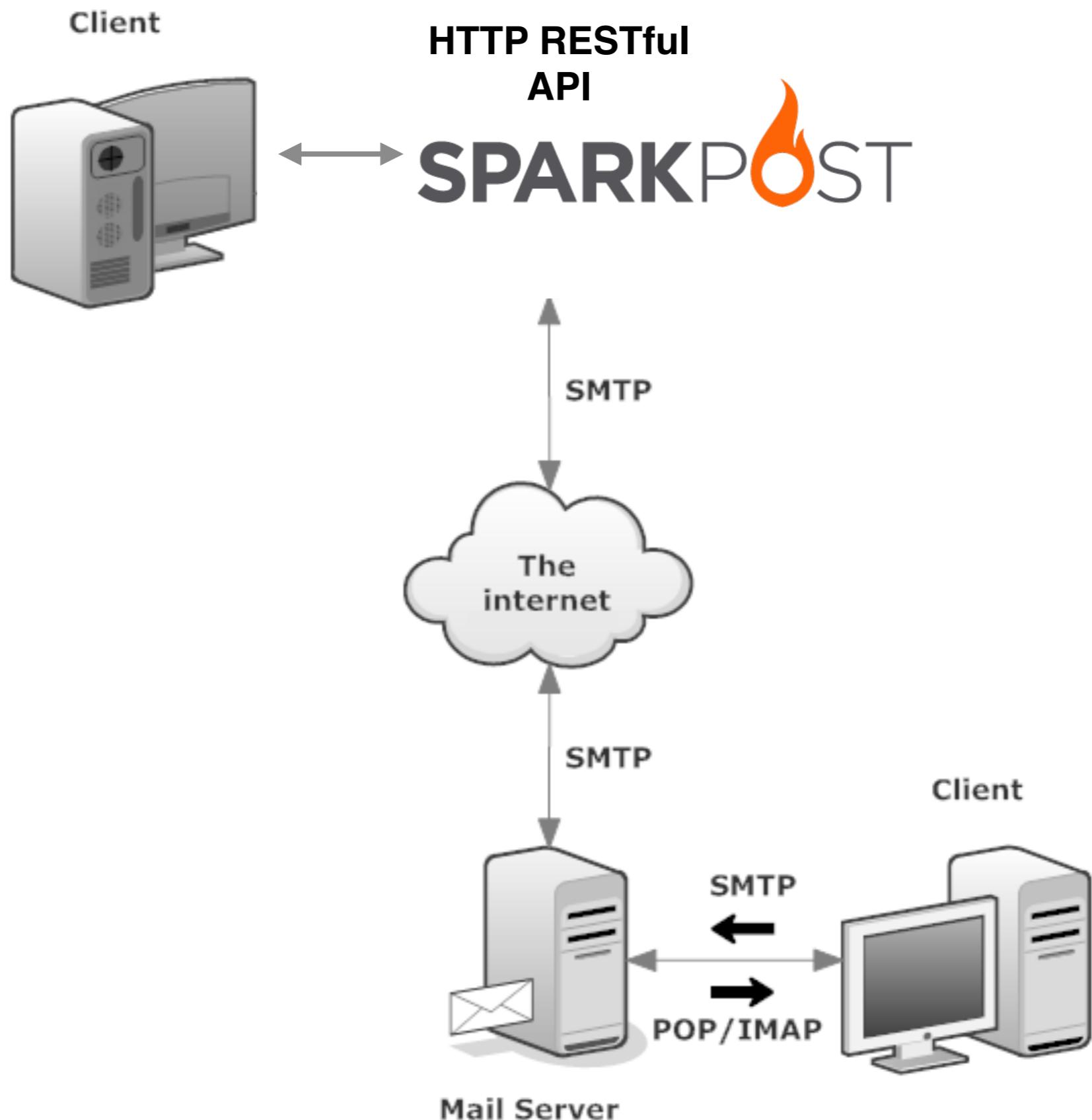


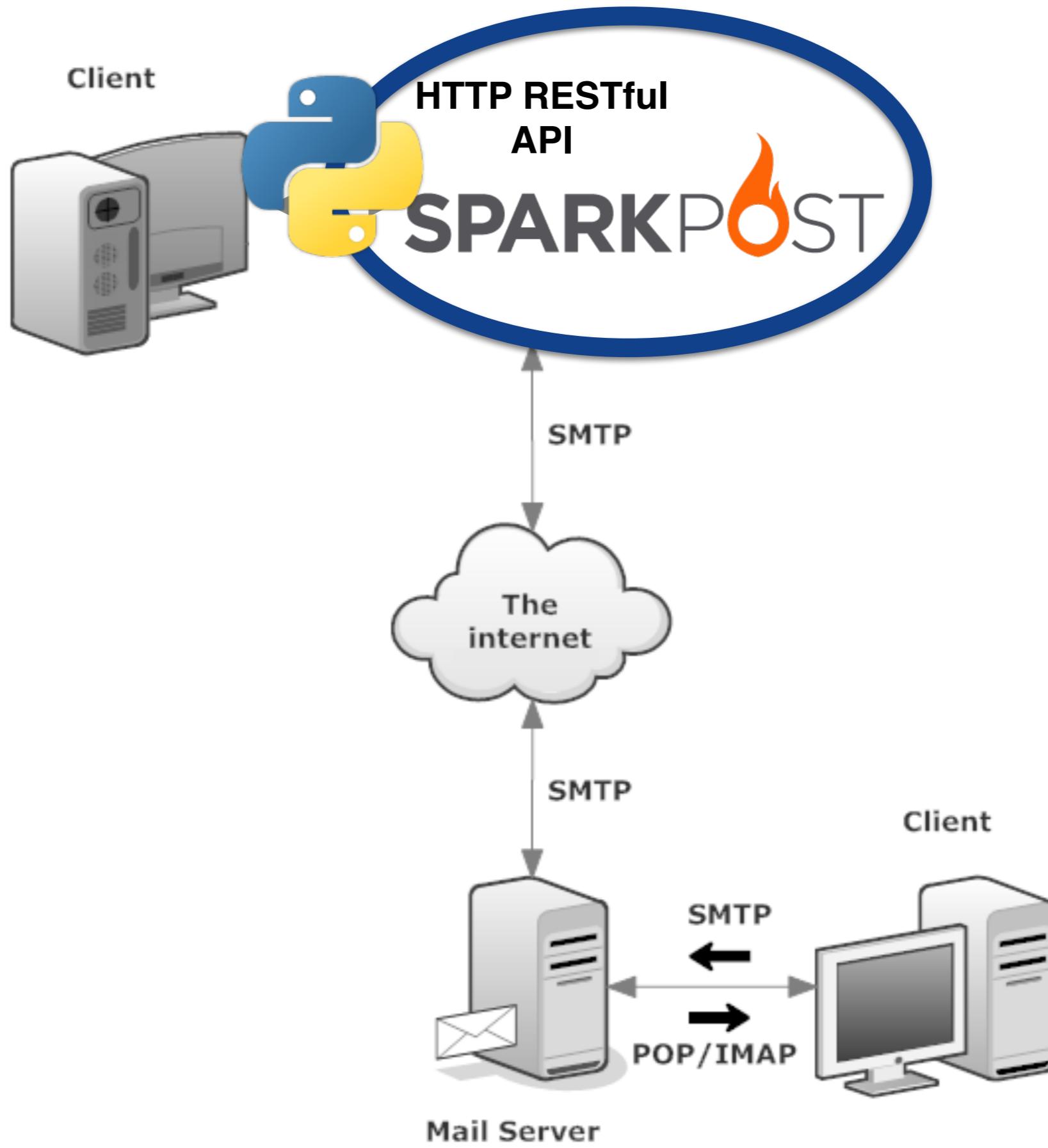


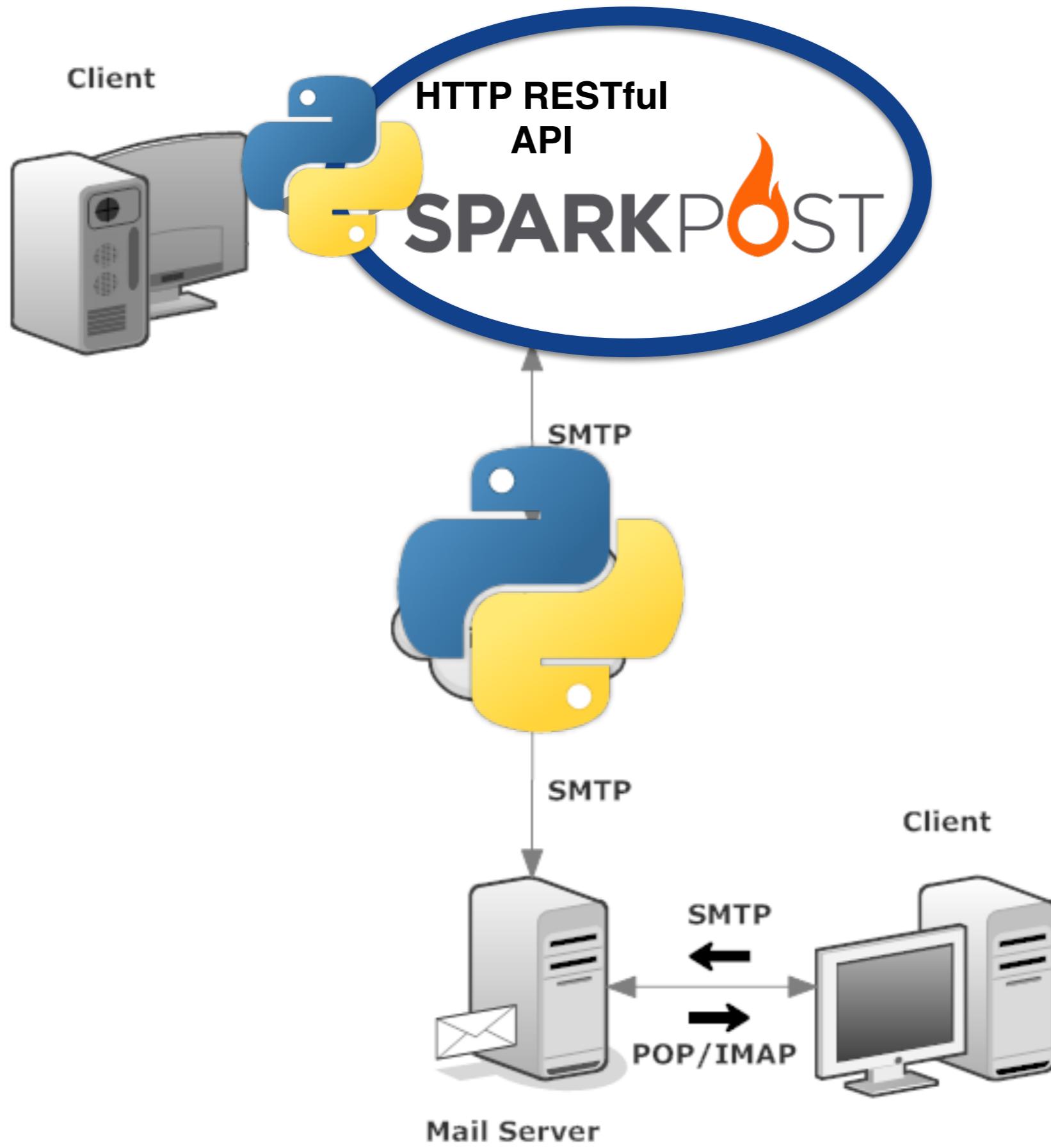
MANDRILL











```
from sparkpost import SparkPost

sp = SparkPost('YOUR_API_KEY')

response = sp.transmissions.send(
    recipients=['sally@company.com'],
    html='<p>Hello, this is my message</p>',
    text='Hello, this is my message',
    from_email='brandon@aubie.ca',
    subject='Hello from python-sparkpost'
)

print(response)
# outputs {u'total_accepted_recipients': 1, u'id': u'47960765679942446', u'total_rejected_recipients': 0}
```

<https://github.com/SparkPost/python-sparkpost>

```
from sparkpost import SparkPost

sp = SparkPost('YOUR_API_KEY')

response = sp.transmissions.send(
    recipients=['sally@company.com'],
    html='<p>Hello, this is my message</p>',
    text='Hello, this is my message',
    from_email='brandon@aubie.ca',
    subject='Hello from python-sparkpost'
)

print(response)
# outputs {u'total_accepted_recipients': 1, u'id': u'47960765679942446', u'total_rejected_recipients': 0}
```

} **HTTP API**

<https://github.com/SparkPost/python-sparkpost>

```
response = sp.transmissions.send(  
    recipients=[  
        'postmaster@example.com',  
        'you@me.com',  
        {  
            'address': {  
                'email': 'john.doe@example.com',  
                'name': 'John Doe'  
            }  
        }  
    ],  
    html='<p>Hello world {{name}}</p>',  
    text='Hello world {{name}}',  
    from_email='test@sparkpostbox.com',  
    subject='Example Script',  
    description='contrived example',  
    custom_headers={  
        'X-CUSTOM-HEADER': 'foo bar'  
    },  
    track.opens=True,  
    track.clicks=True,  
    start_time='2015-11-06T09:10:00-05:00',  
    campaign='python-sparkpost example',  
    metadata={  
        'key': 'value',  
        'arbitrary': 'values'  
    },  
    substitution_data={  
        'name': 'Example User'  
    }  
)
```

```
response = sp.transmissions.send(  
    recipients=[  
        'postmaster@example.com',  
        'you@me.com',  
        {  
            'address': {  
                'email': 'john.doe@example.com',  
                'name': 'John Doe'  
            }  
        }  
    ],  
    html='<p>Hello world {{name}}</p>',  
    text='Hello world {{name}}',  
    from_email='test@sparkpostbox.com',  
    subject='Example Script',  
    description='contrived example',  
    custom_headers={  
        'X-CUSTOM-HEADER': 'foo bar'  
    },  
    track.opens=True,  
    track.clicks=True,  
    start_time='2015-11-06T09:10:00-05:00',  
    campaign='python-sparkpost example',  
    metadata={  
        'key': 'value',  
        'arbitrary': 'values'  
    },  
    substitution_data={  
        'name': 'Example User'  
    }  
)
```

} Templates



SPARKPOST



REPORTS



LISTS



TEMPLATES



ACCOUNT



HELP & DOCS



SIGN OUT

Templates | Reservation Confirmation User : Draft

[SAVE](#)[SAVE & PUBLISH](#)[VIEW PUBLISHED](#)

Template Name *

ID *

A Unique ID for your template, we'll fill this in for you.

From Name

A friendly from for your recipients.

From Email *

Choose Domain...

Reply To:

An email address to reply to.

Subject *

Description

Template Content *

[HTML ▾](#)[TEST SUBSTITUTION DATA](#)[PREVIEW & SEND](#)

```
1 {{if developer_redirect }}  
2 <div style="padding: 15px 5px; background-color: #fff; border-bottom: 1px solid #004864; margin-bottom: 10px;">  
3   <strong>Developer  
4   {{end}}</strong>  
5   <div style="width: 560px; margin-left: auto; margin-right: auto;">  
6     <div style="background-color: #fff; padding: 5px; border-bottom: 1px solid #004864; text-align: center;">  
7         
8     </div>  
9   </div>  
10  <div style="padding: 10px; color: #000000; border-bottom: 1px solid #ED217C;">  
11    <h2 style="margin-bottom: 0; color: #00487d; font-family: arial, sans-serif;">Reservation Confirmation</h2>  
12    <p style="margin-top: 0; font-family: arial, sans-serif; font-size: smaller;">Last modified: {{last_modified}}</p>  
13    <table style="font-family: arial; margin-left: auto; margin-right: auto;" cellpadding="0" cellspacing="0">  
14      <tr>  
15        <td style="font-size: bold; font-size: larger; text-align: center; width: 150px;">  
16          {{if loantoken}}  
17            <tr>  
18              <td style="font-size: bold; font-size: larger; text-align: center; width: 150px;">  
19                <a href="https://app.qreserve.com/#/endearly/{{loantoken}}" style="text-decoration: none; color: #004864; font-weight: bold;">{{loantoken}}</a>  
20              </td>  
21            </tr>  
22          {{end}}  
23        </td>  
24      </tr>  
25      <tr>  
26        <td style="background-color: #00486d; color: #FFFFFF; font-weight: bold; width: 150px;">  
27          {{user}}</td>  
28      </tr>  
29      <tr>  
30        <td style="background-color: #ED217C; color: #FFFFFF; font-weight: bold; width: 150px;">  
31          Approval Status: {{status}}</td>  
32      </tr>  
33    </table>  
34  </div>  
35  </div>
```

 **QReserve** to me 

9:43 AM

Q·RESERVE

Reservation Confirmation

Last modified by **Brandon Aubie**.

Brandon Aubie

Approval Status: Approved

Calendar Resource

Brandon's Test Facility

Purpose: Demonstrate email templates.

START DATE

Wednesday, July 06, 2016 at 10:00AM EDT

END DATE

Wednesday, July 06, 2016 at 11:00AM EDT

TOTAL DURATION

1.00 hours

Estimated Cost: **€5.00** (€5.00 per hour) *

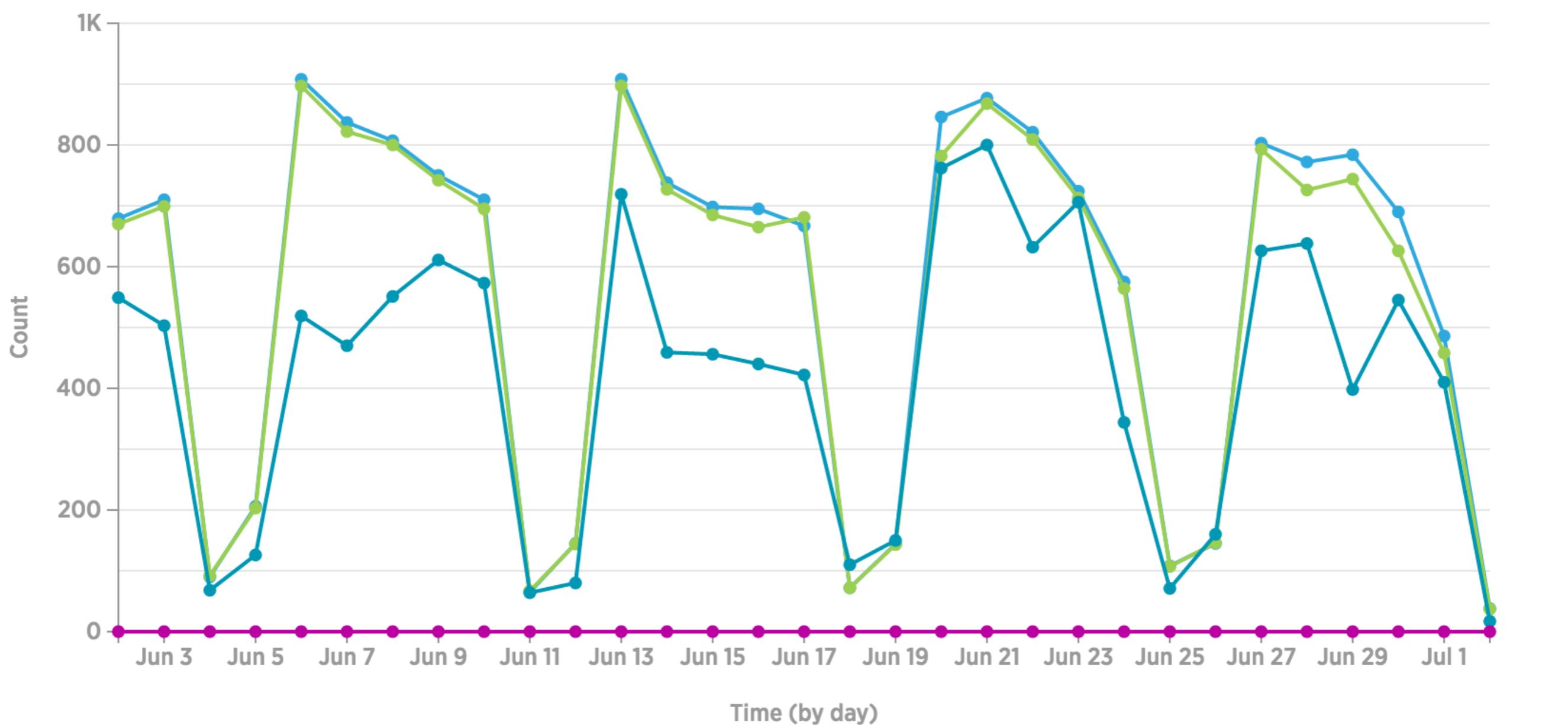
* This is not an invoice. Actual price may vary.

Date --
Rendered --

Targeted --
Clicks --

Accepted --

LINEAR LOG POWER



Delayed Reasons

Reason	Domain	Delayed ↓	Delayed First Attempt	Delayed Rate
454 4.4.4 [internal] no MX or A for domain	uandes.cl	457	64	0.37%
421 4.4.0 [internal] no MXs for this domain could be reached at this time	uandes.cl	238	34	0.2%
451 Message Tempfailed: http://spambusters.mail.umich.edu/troubleshoot	umich.edu	199	193	1.13%
451 4.2.1 Mailbox busy, try again later	yorku.ca	151	149	0.87%
454 4.4.4 [internal] no MX or A for domain	regenero.cl	141	21	0.12%
454 4.4.4 [internal] no MX or A for domain	miuandes.cl	60	9	0.05%

Sending Domains

 NEW DOMAIN

 **qreserve.com**

Ready to send.

 EDIT 

 **SPF Text Record**

 **DKIM Record**

Verify by Email

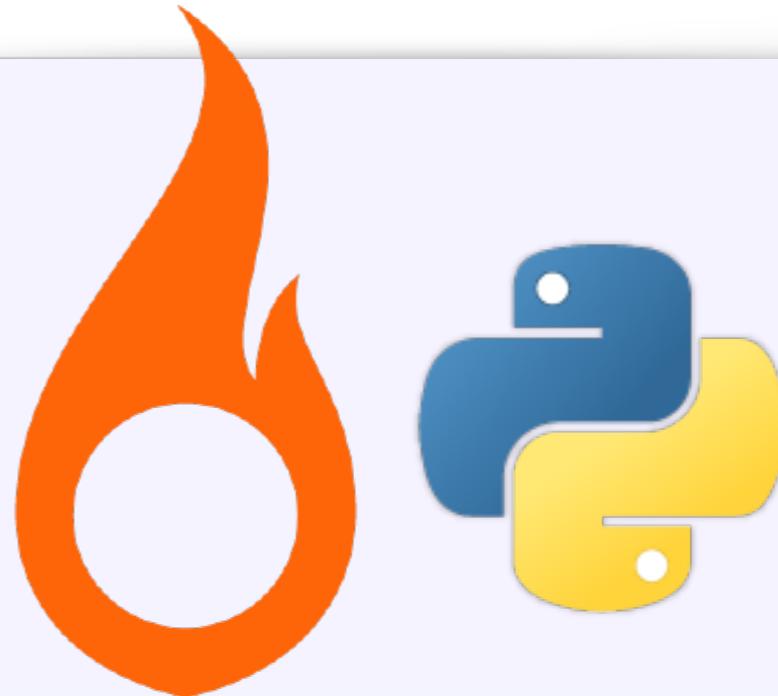
[Email Options](#)

```
from sparkpost import SparkPost

sp = SparkPost('YOUR_API_KEY')

response = sp.transmissions.send(
    recipients=['sally@company.com'],
    html='<p>Hello, this is my message</p>',
    text='Hello, this is my message',
    from_email='brandon@aubie.ca',
    subject='Hello from python-sparkpost'
)

print(response)
# outputs {u'total_accepted_recipients': 1, u'id': u'47960765679942446', u'total_rejected_recipients': 0}
```



- Get all of the advanced SPAM mitigation tools for free.
- Get a trusted SMTP server that does the right thing.
- Don't worry about slow or misbehaving SMTP servers.
- Convenient reporting and template tools.
- Just worry about your emails, not the in-between.

That's how SparkPost in Python works.

*...but never do it like that in production,
of course.*



Q.



Be Safe. Wrap it up.

To send an email in QReserve, we call:

```
from mailer import send_template  
...  
send_template([reservation.user.email], ...)  
...
```

99% of our code doesn't know SparkPost exists, which made it far easier to switch from Mandrill to SparkPost earlier this year. We only had to change one function.

Trust No One

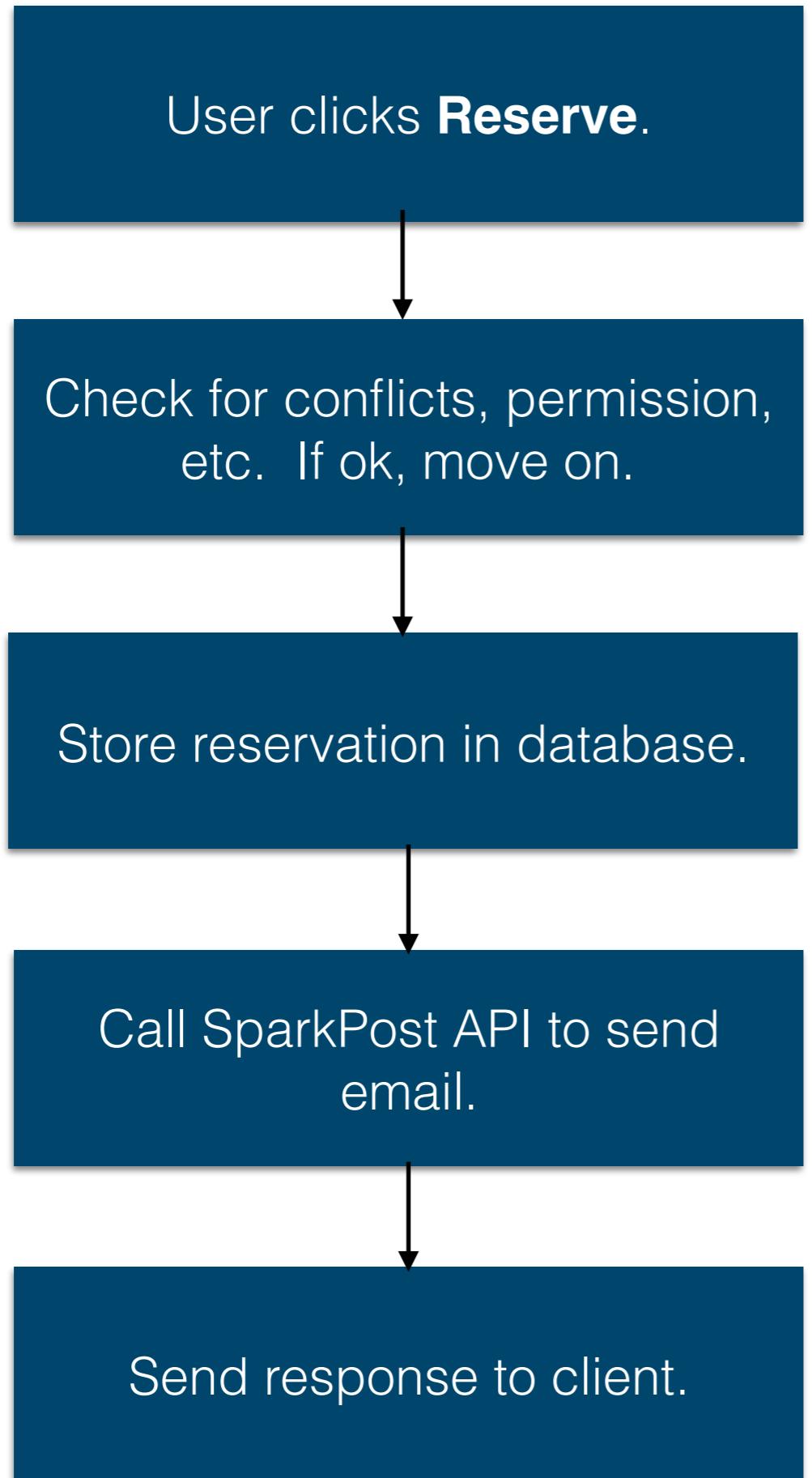
SparkPost has \$65M in funding and runs emails for several very large customers, including LinkedIn. They claim to have the most reliable and scalable email infrastructure of any company.

Trust No One

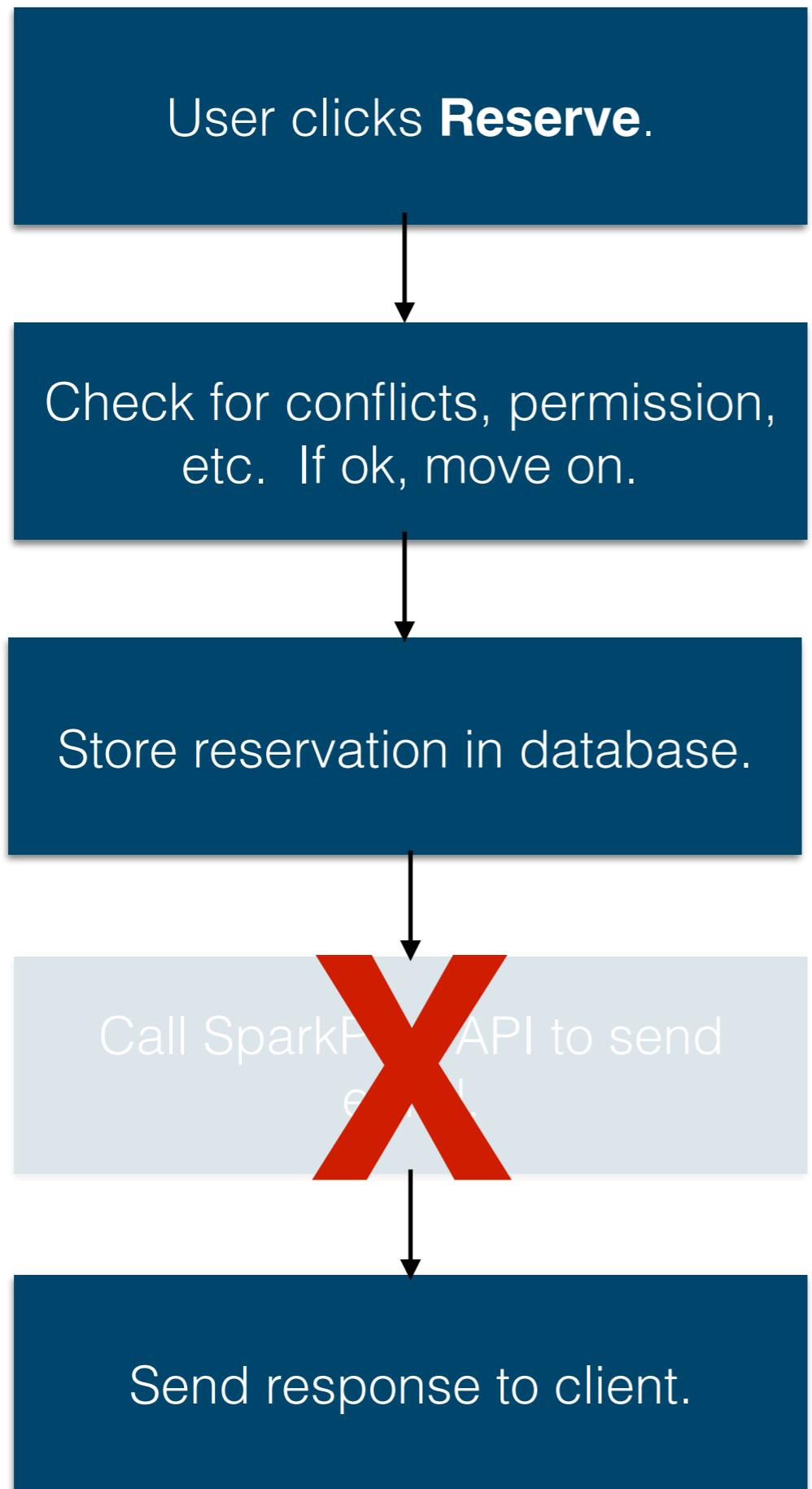
SparkPost has \$65M in funding and runs emails for several very large customers, including LinkedIn. They claim to have the most reliable and scalable email infrastructure of any company.

On May 11th, 2016, their transmissions API started returning 504 Gateway Timeout responses.

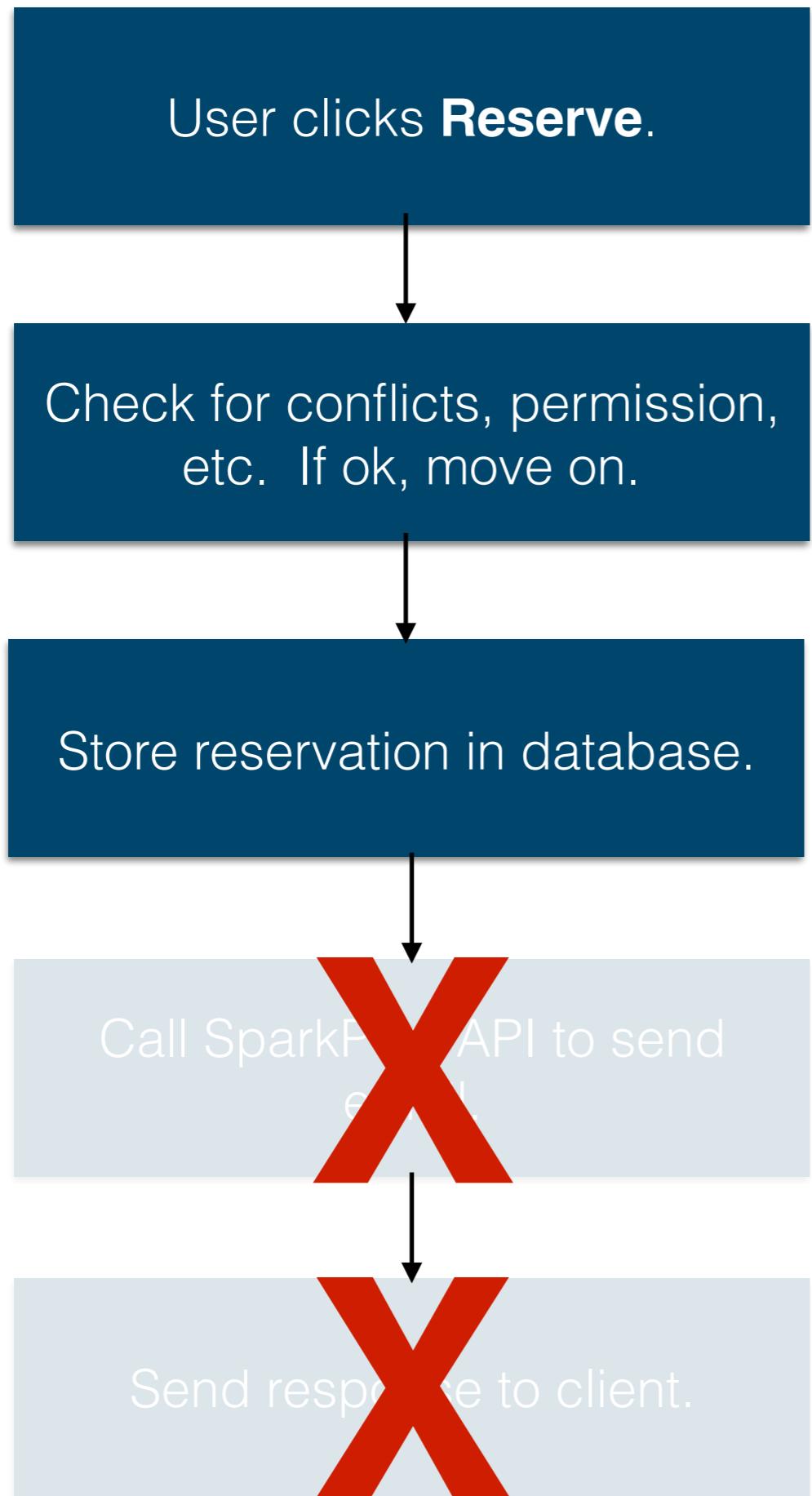
Before



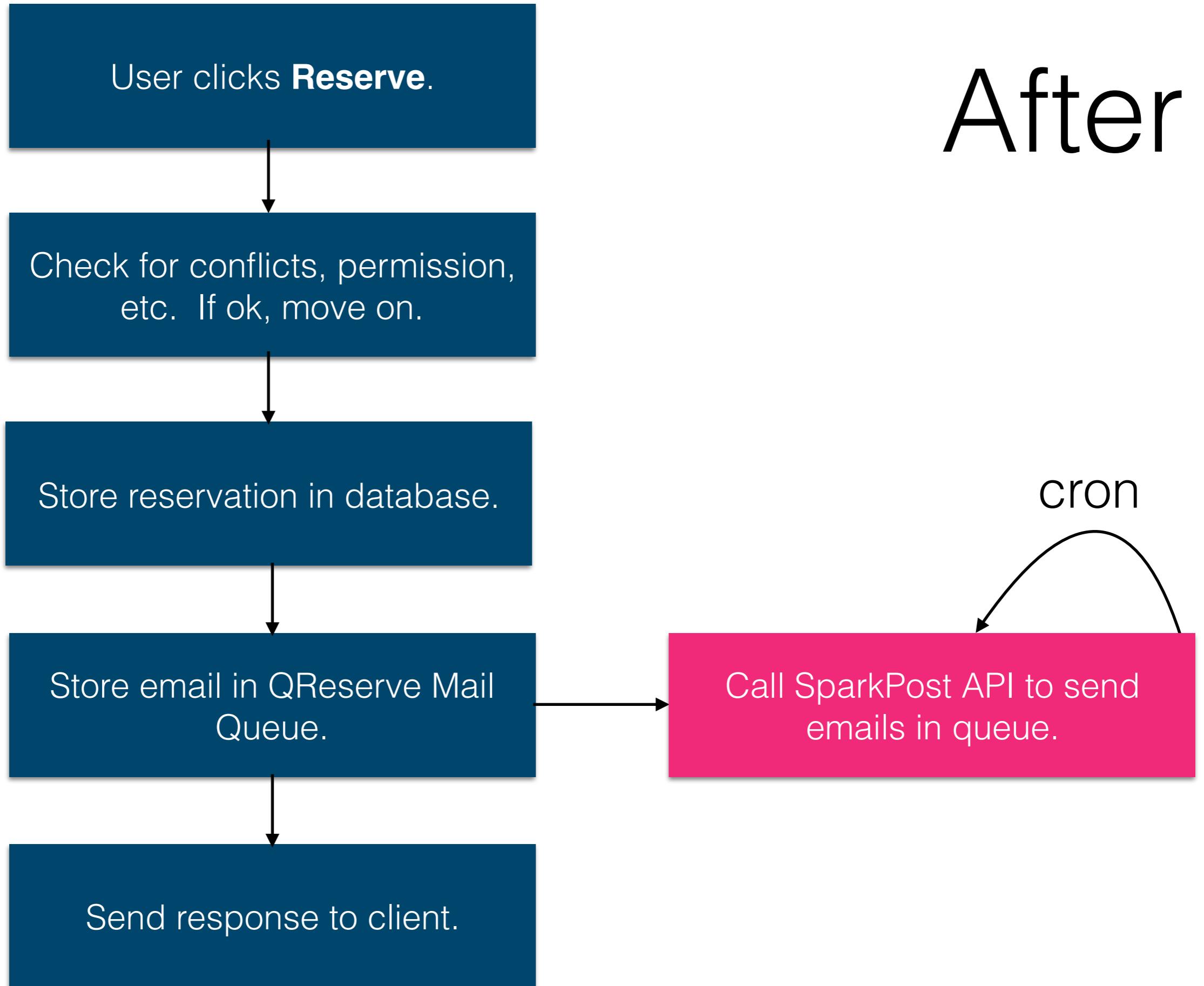
Before



Before



After



You're not done yet

- Email verification emails. I recommend **itsdangerous** for tokens.
- Unsubscribed address management.
- Customer support. Some people simply will never receive your email and you'll never know why.
- HTML email. Remember table layouts? Yeah...

www.aubie.ca

brandon@aubie.ca

[@brandonaubie](https://twitter.com/brandonaubie)

[LinkedIn](https://www.linkedin.com/in/brandonaubie/)