CS210 Student Guide: Verified Email Setup with GitHub, Namecheap, and Brevo

What You'll Need

- A GitHub account
- A free `.me` domain via GitHub Student Developer Pack
- A Namecheap account (created during domain claim)
- A Brevo account (brevo.com)
- Basic Python skills to use `mailer.py`

Step 1: Claim Your Free `.me` Domain

1. Go to https://nc.me/landing/github
2. Scroll to **"Namecheap offer FAQ"**
3. Click **"Get started right here."**
4. Search for a domain ending in `.me` (e.g., `cs210studentname.me`)
5. Sign in with GitHub and authorize Namecheap
6. Complete the free checkout — Namecheap will email your login info

Step 2: Create Two GitHub Repositories

Repo Name	Purpose
`cs210-root-verify`	Verifies `cs210studentname.me`
`cs210-mail-verify`	Verifies `mail.cs210studentname.me`

In each repo, add a simple `index.html`:

```
<!DOCTYPE html>
<html>
  <head><title>Verified</title></head>
  <body><h1>This domain is verified</h1></body>
</html>
```

Step 3: Enable GitHub Pages

For each repo:

1. Go to **Settings > Pages**

2. Under **Source**, select `main` branch and `/root`
3. GitHub will generate a public URL

Step 4: Point Your Domain to GitHub Pages

In **Namecheap > Domain List > Manage > Advanced DNS**, add these records:

Type	Host	Value	TTL	Purpose
CNAME	`@`	`yourusername.github.io`	Automatic	Root domain (`cs210studentname.me`)
CNAME	`mail`	`yourusername.github.io`	Automatic	Subdomain (`mail.cs210studentname.me`)

Replace `yourusername` with your actual GitHub username.
Do not include `https://` or trailing slashes.

Step 5: Create a Brevo Account and Add Your Domain

1. Go to brevo.com and sign up
2. Go to **Settings > Senders & Domains > Domains**
3. Click **Add a Domain**
4. Enter your full domain (e.g., `cs210studentname.me`)
5. Brevo will generate DNS records for verification

Step 6: Add Brevo DNS Records in Namecheap

In **Advanced DNS**, add the following records exactly as shown:

Type	Host	Value	TTL	Purpose
TXT	`@`	`brevo-code:4e2b86039e6ddd1892eb0fd3a660d8ce`	Automatic	Brevo domain verification
CNAME	`brevo1._domainkey`	`b1.cs210studentname-me.dkim.brevo.com.`	30 min	DKIM record 1
CNAME	`brevo2._domainkey`	`b2.cs210studentname-me.dkim.brevo.com.`	30 min	DKIM record 2
TXT	`_dmarc`	`v=DMARC1; p=none; rua=mailto:rua@dmarc.brevo.com`	Automatic	DMARC policy

Replace `cs210studentname` with your actual domain name prefix.
The hyphen (`-me`) is part of Brevo's formatting and is correct.

Step 7: Verify Your Domain in Brevo

After adding the records:

1. Return to Brevo’s **Domains** section
2. Click **Verify** next to your domain
3. Wait for DNS propagation (usually under 1 hour)

Step 8: Get Your Brevo API Key

1. Go to **SMTP & API > API Keys**
2. Click **Create a New API Key**
3. Name it (e.g., `cs210-mailer`)
4. Choose **v3 (recommended)**
5. Click **Generate** and **copy the key immediately**

Step 9: Send Email with `mailer.py`

Here’s a basic Python script using Brevo’s HTTP API:

```
import requests

headers = {
    "api-key": "your_api_key_here",
    "Content-Type": "application/json"
}

data = {
    "sender": {"name": "CS210", "email": "your_verified@cs210studentname.me"},
    "to": [{"email": "student@example.com"}],
    "subject": "Welcome to CS210",
    "htmlContent": "<h1>Hello from CS210!</h1>"
}

response = requests.post("https://api.brevo.com/v3/smtp/email", headers=headers,
json=data)
print(response.status_code, response.text)
```

Troubleshooting Tips

- DNS changes can take up to 24 hours
- Use whatsmydns.net to check TXT and CNAME propagation
- Double-check spelling and spacing in DNS records
- GitHub Pages may take a few minutes to issue SSL

- Brevo won’t verify until all records are correct and visible

NOTES

Step 6 DNS Records Breakdown

Host	Type	Purpose	Protocol
`@`	TXT	Brevo domain verification	SPF (custom Brevo code)
`brevo1._domainkey`	CNAME	DKIM signature key 1	DKIM
`brevo2._domainkey`	CNAME	DKIM signature key 2	DKIM
`_dmarc`	TXT	DMARC policy and reporting	DMARC

- SPF: Brevo uses a TXT record with a custom code (`brevo-code:...`) to verify domain ownership — this is not a traditional SPF record, but it plays a similar role in confirming sender legitimacy.
- DKIM: These two CNAME records point to Brevo’s public keys for signing outgoing emails.
- DMARC: This TXT record defines how receiving servers should handle failed SPF/DKIM checks and where to send reports.

What Are SPF, DKIM, and DMARC?

These are email authentication protocols that help prevent spoofing, phishing, and spam. They tell receiving mail servers: “This email is legit — it really came from us.”

SPF — *Sender Policy Framework*

- What it does: Lists which servers are allowed to send email on behalf of your domain.
- How it works: When an email arrives, the recipient checks your domain’s SPF record to see if the sending server is authorized.
- Example: If Brevo sends email for `cs210studentname.me`, your SPF record says: “Yes, Brevo is allowed.”

DKIM — *DomainKeys Identified Mail*

- What it does: Adds a digital signature to each email using a private key.
- How it works: The recipient uses your public DKIM key (stored in DNS) to verify that the email wasn’t altered in transit.
- Example: Brevo signs your email with a private key; the recipient checks the signature using your DKIM record.

DMARC — *Domain-based Message Authentication, Reporting & Conformance*

- What it does: Tells receiving servers what to do if SPF or DKIM checks fail.

- How it works: You publish a DMARC policy in DNS (e.g., ”p=none” means don’t reject, just report failures).
- Example: If someone tries to spoof your domain, DMARC can instruct servers to reject or quarantine the message — and send you a report.

Email Authentication Analogy: The CS210 Club

Imagine your email system is like a private CS210 club. Only trusted members are allowed to send messages on behalf of the club. Here's how SPF, DKIM, and DMARC work in that world:

SPF = Guest List at the Door

- Think of SPF as the guest list.
- When someone shows up claiming to be from CS210, the bouncer checks:
  “Is this person on the list of approved senders?”
- If they’re not on the list, they’re turned away or flagged.

In email terms: SPF checks if the sending server is allowed to send mail for your domain.

DKIM = Secret Handshake

- DKIM is like a secret handshake that only real CS210 members know.
- When a message arrives, the recipient checks:
  “Did this person use the correct handshake?”
- If the handshake matches, the message is trusted.

In email terms: DKIM uses a digital signature to prove the message wasn’t tampered with.

DMARC = Security Policy

- DMARC is the security policy posted at the club entrance.
- It tells the bouncer what to do if someone fails the guest list or handshake test:
  - Let them in anyway? (p=none)
  - Warn the manager? (send a report)
  - Kick them out? (p=reject)

In email terms: DMARC tells receiving servers how to handle suspicious messages and where to send alerts.

Why It Matters

Without these checks:

- Anyone can impersonate your domain
- Your emails might land in spam
- You lose credibility with mail servers

With them:

- You prove you're legit
- You protect your domain
- You increase deliverability