

API Keys

Outline for Today

- ***Getting Data Sets***
  - There's a lot of useful stuff out there!
- ***Web APIs***
  - Controlling software via the web.
- ***API Keys***
  - A username and password in one.
- ***Securing API Keys***
  - A necessary and tricky step.

Visualizing Data Sets



https://www.eia.gov

What is an API?

Web APIs

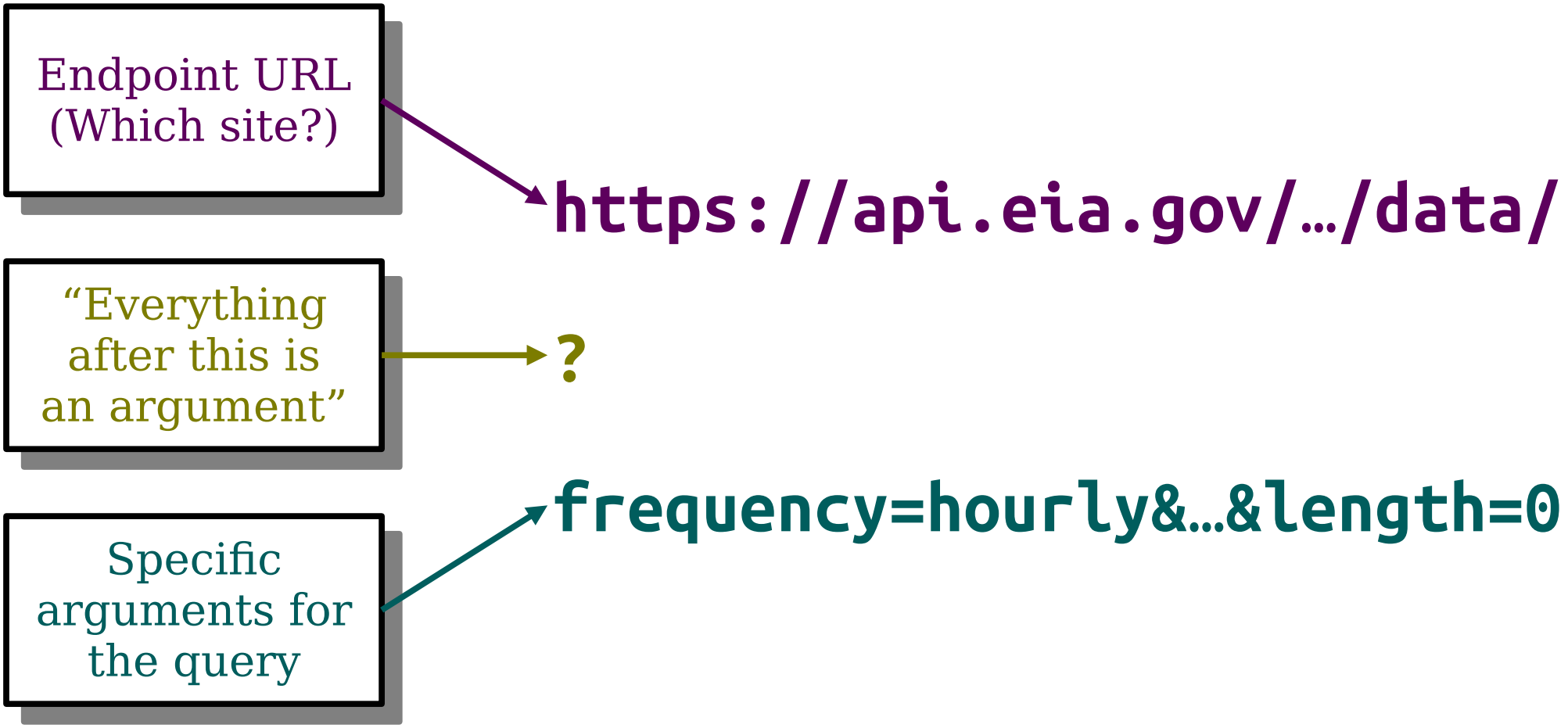
- A ***web API*** (***A***pplication ***P***rogramming ***I***nterface) is a web server designed to let other programs perform tasks automatically.
- For example:
  - `eia.gov`: Query US power generation.
  - `511.org`: Query Bay Area traffic / public transit.
  - Instagram: Publish content, search for mentions, …
  - Stripe: Perform a transaction with a customer, …
  - Robinhood: Trade cryptocurrency. (But… why?)
- Search around online and you'll find *thousands* of these.

```
https://api.eia.gov/v2/
electricity/rto/
fuel-type-data/
data/?frequency=hourly&
data[0]=value&
sort[0][column]=period&
sort[0][direction]=desc&
offset=0&length=5000
```

https://api.eia.gov/…/data/

?

frequency=hourly&…&length=0


Endpoint URL
(Which site?)
https://api.eia.gov/…/data/
“Everything
after this is
an argument”
?
Specific
arguments for
the query
frequency=hourly&…&length=0

API Keys

Economic Realities

- It’s great when governments and businesses share data. But...
  - ... gathering data is expensive.
  - ... running servers under load is expensive.
  - ... large data sets are an economic moat.
- ***Consequence:*** Providers often restrict access to their APIs.
  - ***Rate-Limiting:*** “You can access this API, but don’t make more than *X* queries in *Y* time.”
  - ***Charge for Access:*** “You’re welcome to my data, but you need to pay me $*X* per *Y* unit of time.”
- But APIs are just URLs on the public internet, so how is this enforced?

API Keys

- An ***API key*** is a string of letters and numbers that identifies who is making an API call.
  - These are sometimes called ***API tokens*** or ***access tokens***; no relation to “token” in the LLM sense.
  - Hmmm… where have you heard of this before?
- Some APIs require you to obtain a key before you can use their service.
- There are different means of getting API keys.
  - Some places give them out for free and just want to know who you are.
  - Some places charge (steep!) fees for API keys.
- Not all websites have APIs. Alas…

Obtaining an `eia.gov` API Key

API Key Security

Web APIs

- A ***web API*** (***A***pplication ***P***rogramming ***I***nterface) is a web server designed to let other programs perform tasks automatically.
- For example:
  - `eia.gov`: Query US power generation.
  - `511.org`: Query Bay Area traffic / public transit.
  - Instagram: Publish content, search for mentions, …
  - Stripe: Perform a transaction with a customer, …
  - Robinhood: Trade cryptocurrency. (But… why?)
- Search around online and you'll find *thousands* of these.

API Key Security

- An API acts like both a username and a password. Anyone who has an API key of yours can perform any actions you can with it.

  ☞ ***You must secure your API keys!*** ☜

- The risks here are real.
  - API key for Instagram: Whoever has it can post content as you.
  - API key for Stripe: Whoever has it can move money out of your account.
  - API key for Robinhood: Whoever has it can control your crypto wallet.
- And, unfortunately, API keys are easily to accidentally leak if you aren't careful.
- How do API keys get leaked in the first place?

Failure Mode 1: ***API Keys on GitHub***

SFGATE

BUSINESS

Uber concealed cyberattack that exposed 57 million people's data

Here's how the hack went down: Two attackers accessed a private GitHub coding site used by Uber software engineers, then used log-in credentials they obtained there to access data stored on an Amazon Web Services account that handled computing tasks for the company. From there, the hackers discovered an archive of rider and driver information. Later, they emailed Uber asking for money, according to the company.

https://www.sfgate.com/business/article/Uber-concealed-cyberattack-that-exposed-57-12375864.php

API Keys on GitHub

- One of *the most common ways* people leak API keys is accidentally uploading them to GitHub.
- It’s really easy to do this accidentally.
  - Programs need API keys to make API calls.
  - API keys are often stored in files with the source code.
  - Files near code get committed and pushed to GitHub.
  - Oops!
- And, if you get this wrong, it’s *really* hard to undo it.
  - By default, git stores *every commit you’ve ever made*. If you accidentally commit API keys even once, it’s really hard to get them out of the commit history.
- And lots of bad actors are constantly looking at GitHub to find leaked keys. 😨

Demo: CanaryTokens on GitHub

How do we prevent this?

.gitignore

- You can create a file named `.gitignore` in a git repository to list files that git should not track or include in commits.
- If you have any files in your repo that contain API keys,

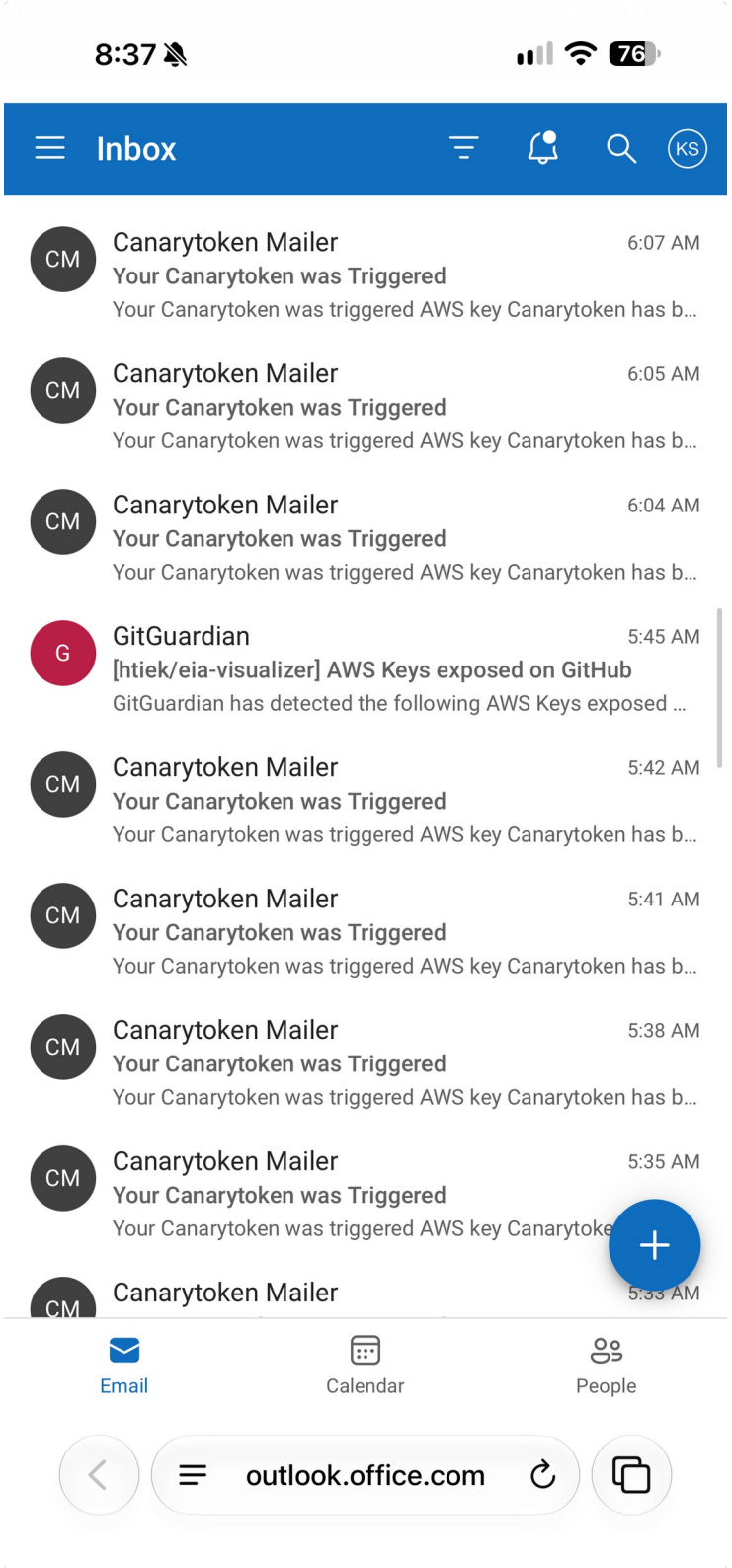
  ☞ ***those files must be .gitignored*** ☜

- Coding agents are getting much better about this, but they aren’t perfect.
- You need to know how to check by hand.

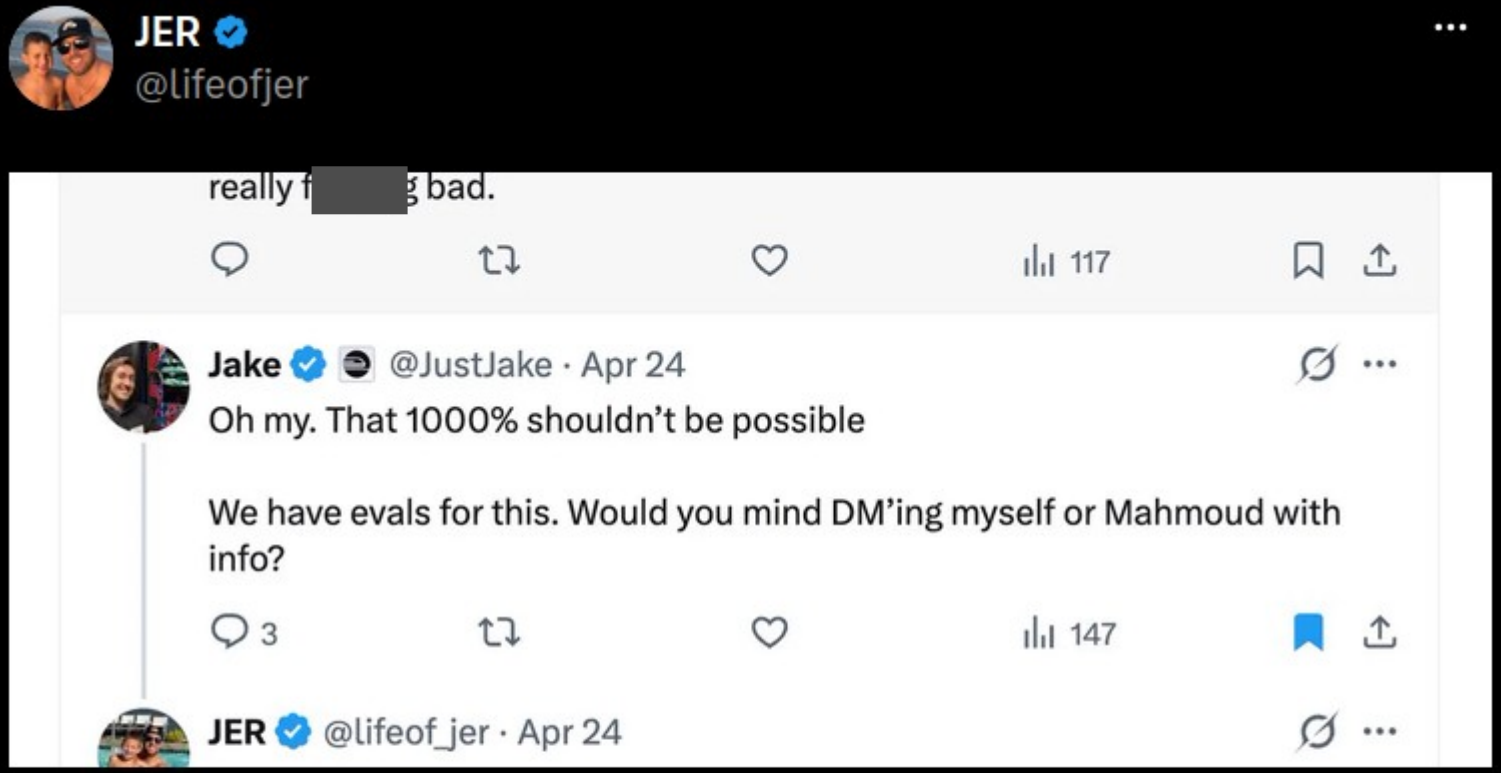
.gitignore

- How to make sure `.gitignore` ignores your keys:
  - Open `.gitignore` and look for something that says `.env.local` or `.env.*`.
  - If it's not there, add it manually yourself.
  - Use `git check-ignore .env.local` to confirm that the file is gitignored. (It should print `.env.local` if everything is good.)
- Yes, your coding agent probably can do this for you, but you don't want to rely on that. It's really bad if keys leak.

How’s our canary token doing?


8:37
Inbox
KS
CM
Canarytoken Mailer
6:07 AM
Your Canarytoken was Triggered
Your Canarytoken was triggered AWS key Canarytoken has b...
CM
Canarytoken Mailer
6:05 AM
Your Canarytoken was Triggered
Your Canarytoken was triggered AWS key Canarytoken has b...
CM
Canarytoken Mailer
6:04 AM
Your Canarytoken was Triggered
Your Canarytoken was triggered AWS key Canarytoken has b...
G
GitGuardian
5:45 AM
[htiek/eia-visualizer] AWS Keys exposed on GitHub
GitGuardian has detected the following AWS Keys exposed ...
CM
Canarytoken Mailer
5:42 AM
Your Canarytoken was Triggered
Your Canarytoken was triggered AWS key Canarytoken has b...
CM
Canarytoken Mailer
5:41 AM
Your Canarytoken was Triggered
Your Canarytoken was triggered AWS key Canarytoken has b...
CM
Canarytoken Mailer
5:38 AM
Your Canarytoken was Triggered
Your Canarytoken was triggered AWS key Canarytoken has b...
CM
Canarytoken Mailer
5:35 AM
Your Canarytoken was Triggered
Your Canarytoken was triggered AWS key Canarytoke
CM
Canarytoken Mailer
5:33 AM
Email
Calendar
People
outlook.office.com

Failure Mode 2: ***Coding Agents***



An AI Agent Just Destroyed Our Production Data. It Confessed in Writing.

1K 1K 5.3K 6.3K

https://x.com/lifeofjer/status/2048103471019434248

Coding Agents and API Keys

- Remember: *anyone* with access to one of your API keys can do whatever they want with it on your behalf.
- This includes coding agents. Thus...

  ☞ ***Keep dangerous keys away from LLMs.*** ☜
- This is trickier than it looks, and the consequences of getting this wrong can be serious!
- How does this happen, and what should you do about it?

Why This is Hard

- You are vibecoding your programs.
- So your agents need access to your source code.
- But you might need an API key file in that directory.
- And coding agents can read any files in your directories.
- What should you do?
  - ***Short answer:*** Don’t use API keys that have any means of inflicting damage until we teach you how to do this correctly.
  - ***Long answer:*** Next Thursday’s lecture. 😀
- I chose the EIA demo *specifically because* that key exists purely for rate-limiting requests to the EIA database. It’s not terrible if it leaks.

Why This is Hard

- An unconstrained coding agent can read *any file* on your computer, even ones you don't intend it to.
- This is a major reason we built you the CS193V Development Environment.
  - It runs in a container that restricts its access to just what's in your `projects/` folder.
  - If you store dangerous keys outside of your `projects/` folder on your machine, coding agents can't get them.
- This is not foolproof.
  - Your coding agent can go online, some folks link their coding agents to their email accounts, etc.
- "A security system is only as strong as its weakest link."

Mitigating the Risk

Mitigating the Risk

- The software engineering community is aware of the risks associated with API keys and is (generally!) trying to reduce it.
- Here are three techniques you should be aware of:
  - Key revocation.
  - Credential rotation.
  - Scoped keys.
- Let’s address these in turn.

Key Revocation

- Suppose you do accidentally leak an API key. What can you do?
- Most key issuers have a mechanism to let you ***revoke*** a leaked key.
  - You tell the issuer “Oops! I leaked my key!”
  - The issuer notes this and disables the key.
  - Anyone with access to the key can no longer use it in conjunction with your account.
- ***Analogy:*** Changing your password after it’s been stolen / leaked.
- If you discover that a key is leaked, or someone tells you that your key is leaked, ***revoke it immediately***! Treat it like your bank password was exposed.


CYBERSECURITY & INFRASTRUCTURE
CISA
SECURITY AGENCY



Lessons Learned from CISA's Recent GitHub Leak

[CISA] has issued a postmortem on a recent data leak in which a contractor published dozens of internal CISA credentials — including AWS Govcloud keys — in a public GitHub repository for almost six months before being notified by KrebsOnSecurity. [...]

https://krebsonsecurity.com/2026/07/lessons-learned-from-cisas-recent-github-leak/

Credential Rotation

- When you sign up for an API key, you can (sometimes) set an expiration date for the key, after which it will no longer work.
- Thus if a key is leaked, there's a limited window where it can be abused.
- This process is called ***credential rotation***.
- Some software companies will rotate API keys given to their developers automatically either daily or hourly.
- ***Analogy:*** Websites that make you change your password every six months.

Scoped Keys

- Many API key providers will give you the ability to decide what operations a generated key will be able to perform.
  - For example: read but not create content on Instagram.
  - For example: read an account balance but not move money or initiate transactions.
- These are called ***scoped keys***.
- If a key gets leaked, the blast radius is confined to just the capabilities associated with the scoped key.
- Where have you seen this before?

Other Approaches

- ***Canary Tokens***
  - We used one today to see just how quickly secrets exposed on GitHub get leaked.
  - Consider adding one in to any file you create with an API key in it. If the file gets leaked and the canary triggers, you know your (real) API key is exposed.
- ***Credential Scanners***
  - There are tools (Gitleaks, TruffleHog, etc.) that automatically scan git repos to look for leaked credentials *before* you deploy to GitHub.
  - Adds an extra layer of security on top of the default.
- You can spend a whole career working on things like this…

A Few Final Thoughts

Be Careful and Have Fun

- API keys let you do amazing things and are essential to modern software.
- I don’t mean to scare you away from using them! You should just understand what the risks are and take appropraite precautions.
- If you need help, you can always come talk to us! We’re happy to help.
- And of course you can talk to your coding agent. Just don’t share keys with it directly. 😀

Next Time

- ***Hosting Websites***
  - Letting everyone see your creations!
- ***Vercel***
  - One hosting provider of many.
- ***Deploying API Keys***
  - Keeping secrets safe while deploying to everyone.