Locking Down Your API

Shawn Wildermuth
MICROSOFT MVP, INSTRUCTOR AND FILMMAKER

@shawnwildermuth https://wilderminds.com

Agenda

Locking Down Your API

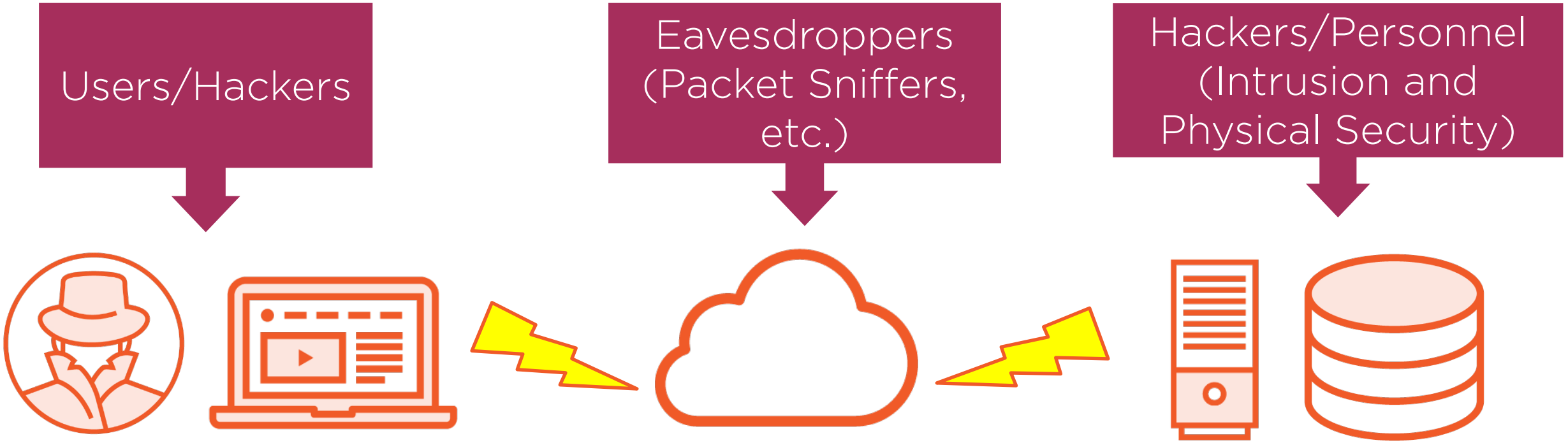
- APIs and Security
- Cross Domain Security
- Authentication/Authorization
- Security considerations during design
- Types of API Security

APIs and Security

Do you need to secure your API?

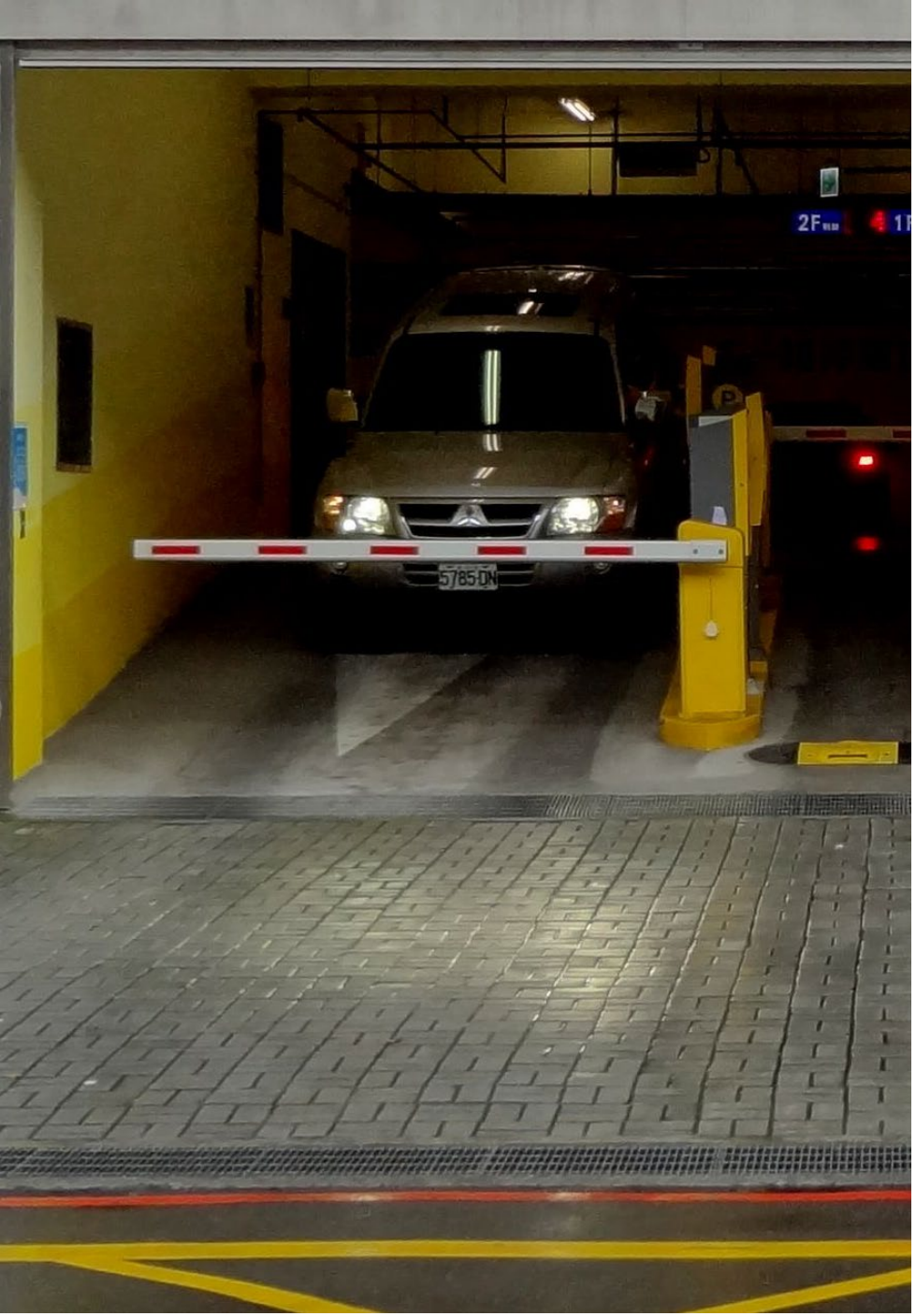
Are you...	Secure?
...using private or personalized data?	Yes.
...sending sensitive data across the 'wire'?	Yes.
...using credentials of any kind?	Yes.
...trying to protect against overuse of your servers?	Yes.

Threats to Your API



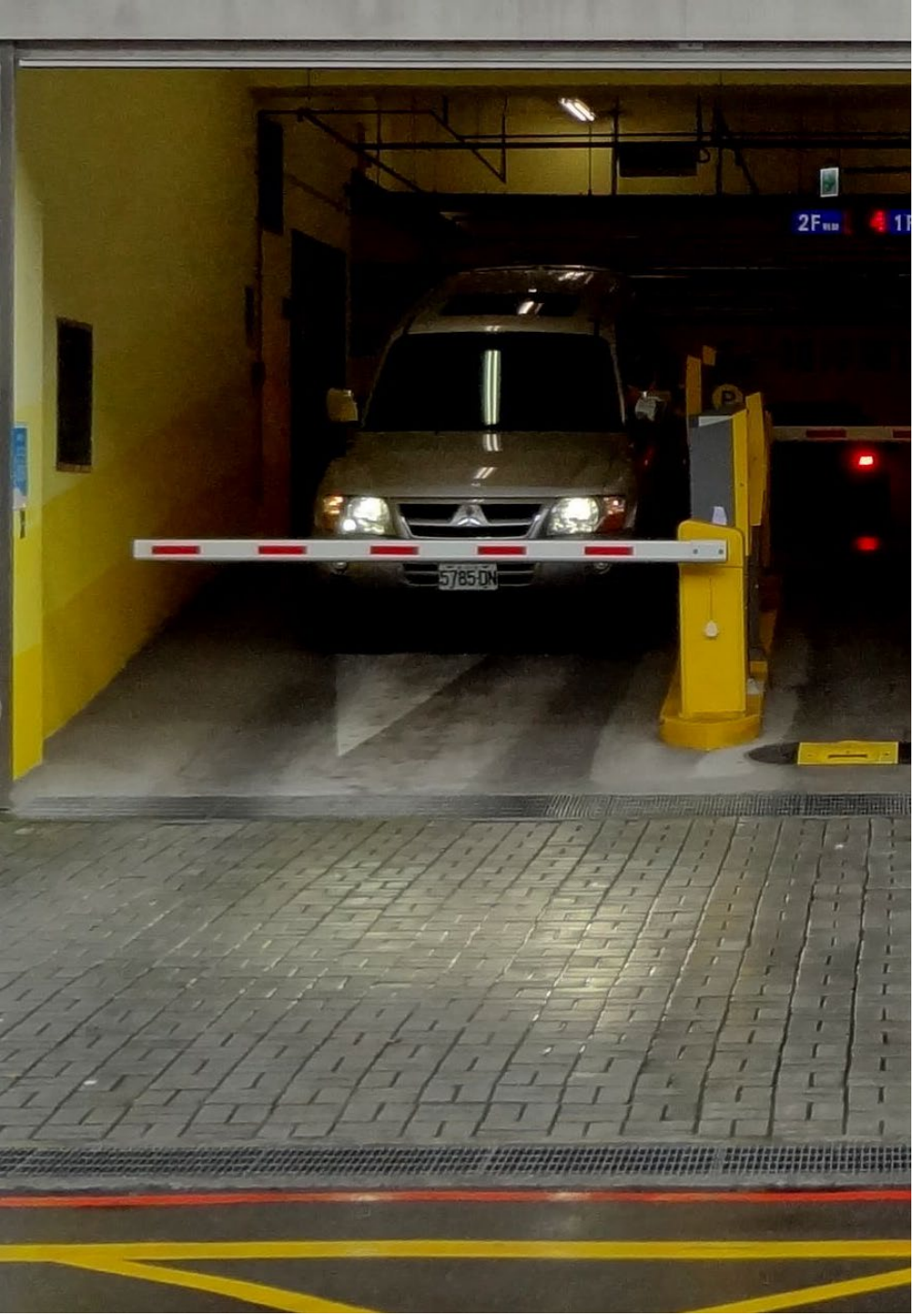
Protect Your API

- Server Infrastructure Security
  - Outside scope of API security
- Secure In-Transit
  - SSL is almost always appropriate
  - Cost of SSL is worth the expense
- Secure the API itself
  - Cross Origin Security
  - Authorization/Authentication



Cross Domain Security

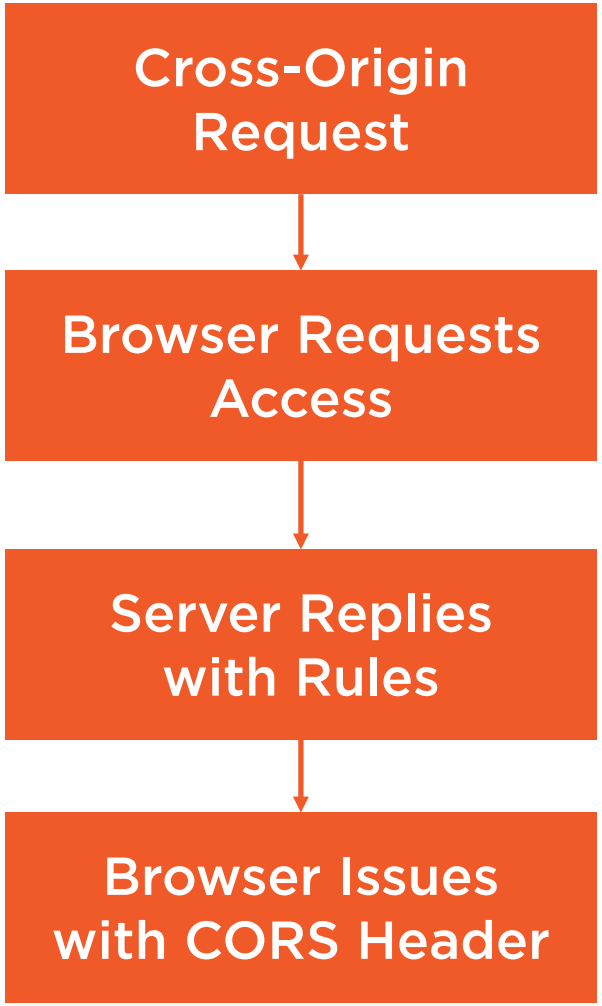
- By default, not allowed
  - But only in the browser
- Public or Private API?
  - Public: Should allow
  - Private: Consider for partners



Cross Origin Resource Sharing (CORS)

- Allows control finely grained control
- Domain, resource, and verb control
- Only limits browser, not app
- Most platforms support CORS

How Does CORS Work?




```
OPTIONS /api/games HTTP/1.1
Origin: http://mysite.com
Access-Control-Request-Method: POST
Host: localhost:8863
```

```
Access-Control-Allow-Methods: GET, POST, OPTIONS
Access-Control-Allow-Origin: http://mysite.com
Content-Length: 0
```

```
POST /api/games HTTP/1.1
Origin: http://mysite.com
Access-Control-Request-Method: POST
Host: localhost:8863
```

Authentication vs. Authorization

Authentication	Authorization
Who you are	What you can do
Information to determine identity	Rules about rights (e.g. Roles, Rights)
Credentials/Claims	

Authentication Types for APIs

- App Authentication
  - Identifying an app for your API
  - Authenticating as the developer!
  - AppID + Key is typical

Authentication Types for APIs

- User Authentication
  - Identifying as a User

Authentication Types for APIs

Cookies

Basic Auth

Token Auth

OAuth

Cookies

- "Cookies are easy, can't I use them"
  - Yes – easiest and common
  - Subject to request forgery
- Depends on your security needs
  - Banks and Pizza Shops aren't equal

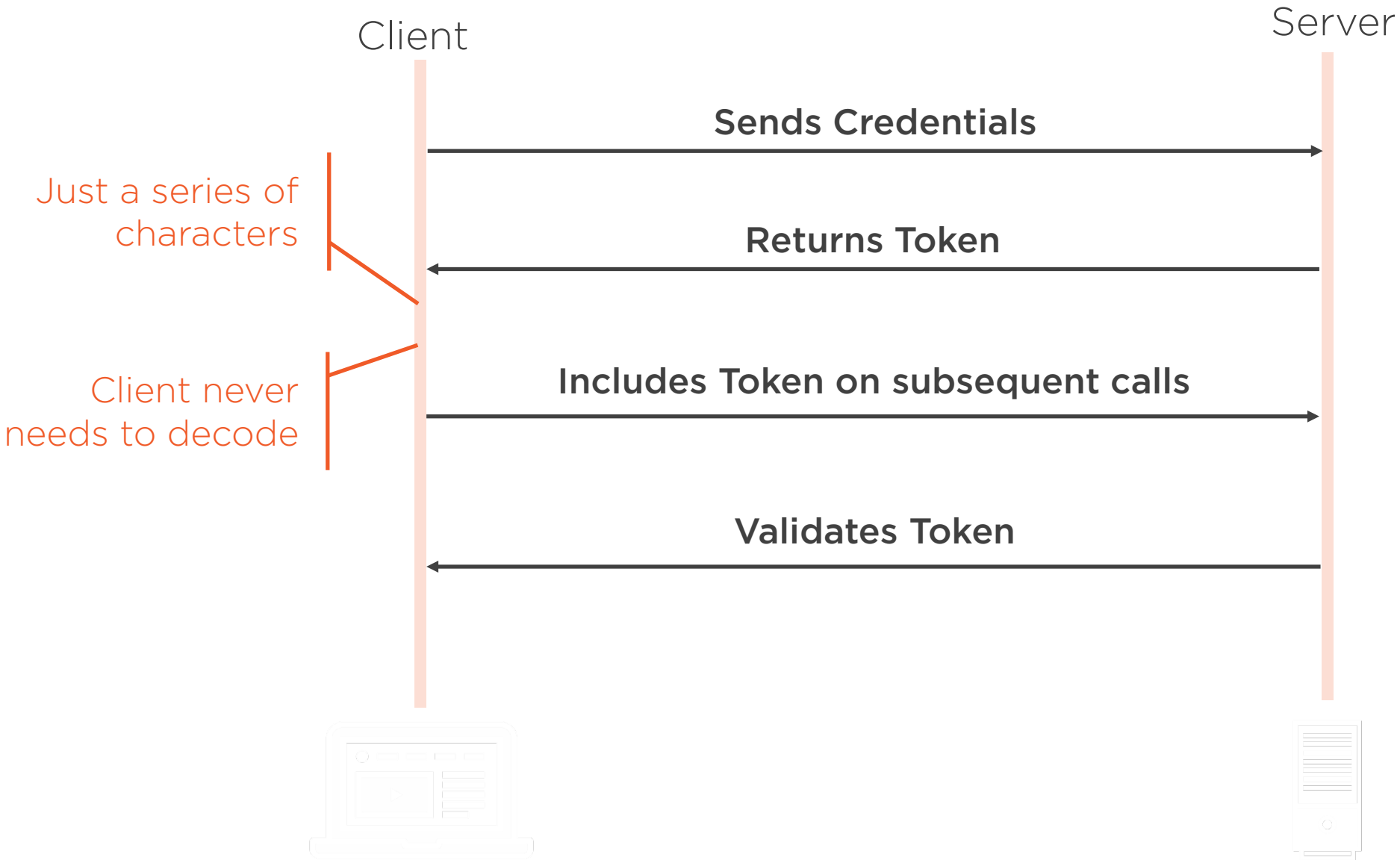
Basic Auth

- Easy to implement
- But not secure, unless enforcing SSL
- Risky still
  - Sends credentials on every request
  - Increases surface area of attacks

Token Based Auth

- Most common
  - Mix of secure and simplicity
- Industry Standard Tokens are easy
- Should expire must faster than cookies
  - Typically 5-20 minutes

Token Authentication



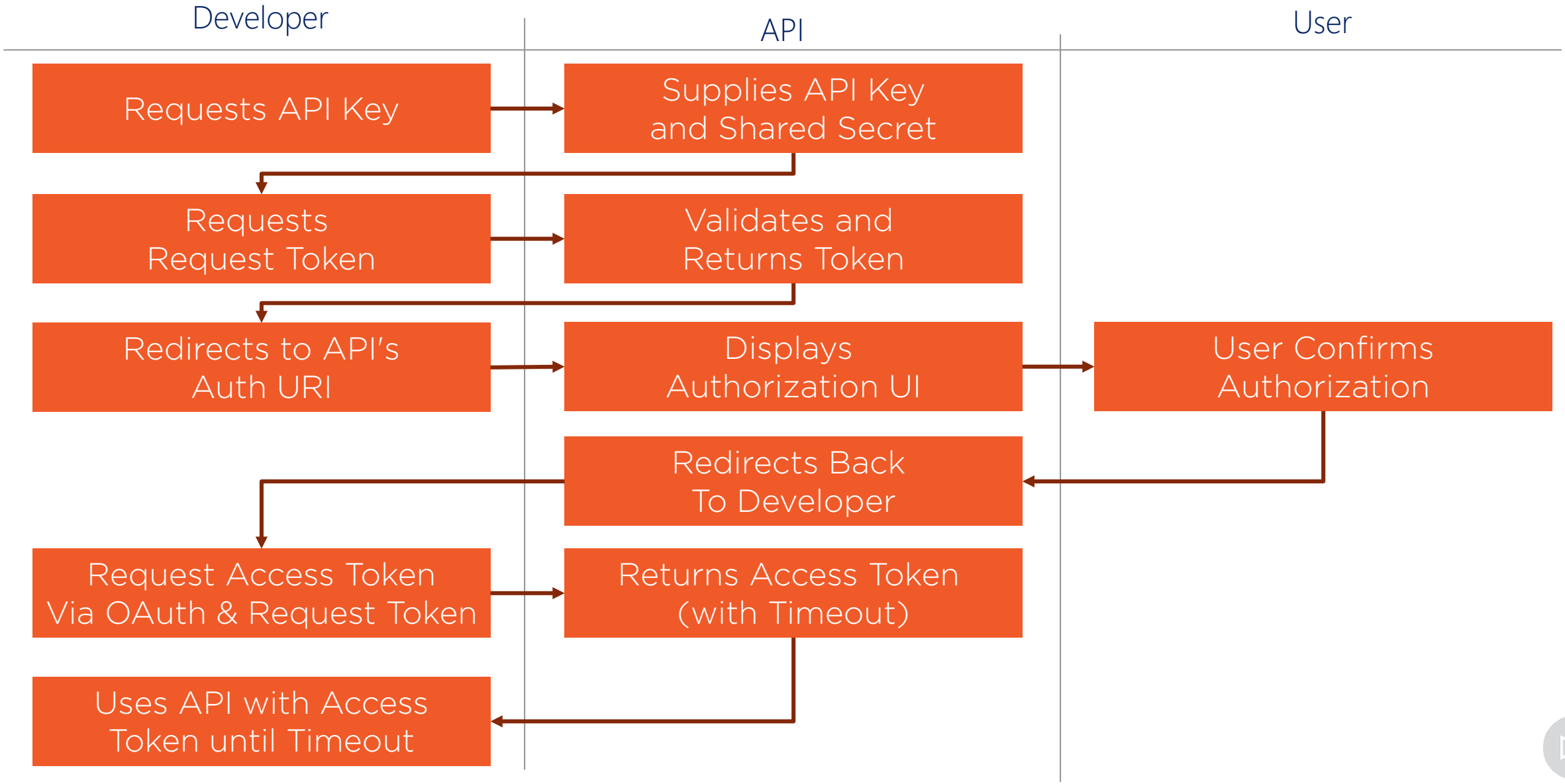
JSON Web Tokens (JWTs)

- Industry standard
- Self-contained, small and complete
  - User Information
  - Claims
  - Validation Signature
  - Other Information

OAuth

- Use trusted third-party to identify
- You never receive credentials
  - User authenticates with third party
  - Use token to confirm identity
  - Safer for you and user

How OAuth Works



OAuth

- If you need this level of security,
  - Don’t implement it by hand please…

What We've Learned

Designing an API without considering security is a big mistake

Security requirements will affect what data you're willing to expose

Be pragmatic with security; don't assume every app needs a vault