Authentication and Authorization in PHP

HTTP AUTHENTICATION

Matthew Setter
PHP SECURITY ENGINEER

@settermjd matthewsetter.com

Module Overview

- What is HTTP authentication?
- Common HTTP authentication methods
- Relevant HTTP headers
- Advantages and disadvantages
- Why HTTPS is essential
- How to implement it in PHP

What is HTTP Authentication?

“An authentication framework, which can be used by a server to challenge a client request, and by a client to provide authentication information, as defined by RFC 7235. It doesn't use login pages, cookies, or session identifiers, rather (it uses) standard fields in the HTTP header.

HTTP Auth (courtesy of MDN)

How Does HTTP Authentication Work?

HTTP Authentication Overview

Quick Recap

- HTTP Basic authentication

HTTP Basic Authentication

Quick Overview

- What is HTTP Basic authentication?
- Advantages and disadvantages
- How to implement it in PHP

HTTP Basic Authentication

- Simplistic authentication process
- Authentication method string is “Basic”
- Credentials are colon-concatenated and base64-encoded

HTTP Basic Authentication Overview

Advantages and Disadvantages

Advantages of Basic Authentication

- Simple to implement
- Quick to deliver
- Passwords can be encrypted
- One round-trip required
- Part of the HTTP specification

Disadvantages of Basic Authentication

- Credentials are not encrypted
- Vulnerable to Man-in-the-middle attacks
  - Essential to use HTTPS/TLS

Disadvantages of Basic Authentication

- How do you logout?
  - Restart your browser
  - Clear your browser's cache
  - Authenticate with incorrect credentials
- Not the most professional user experience

Quick Recap

- What is HTTP Basic authentication?
- Advantages and disadvantages
- Implemented it in PHP

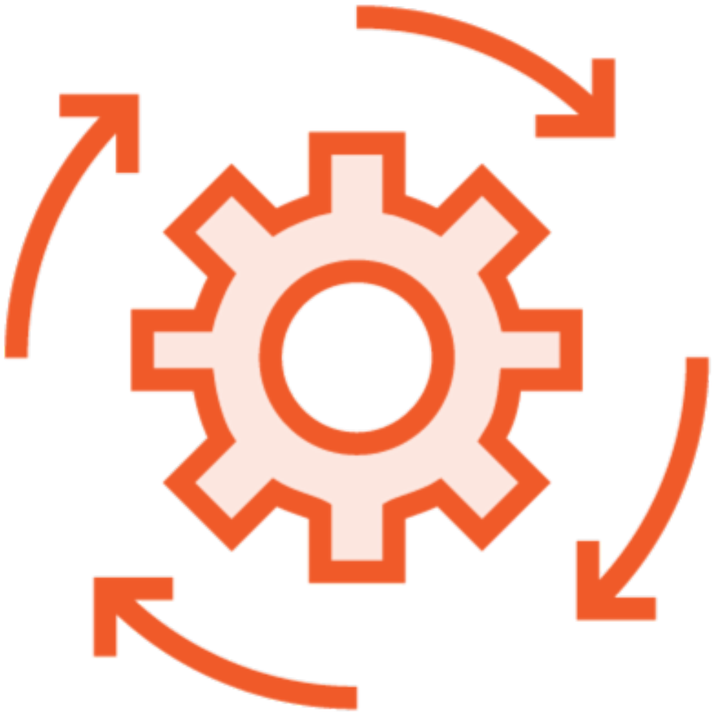
Up Next:
HTTP Digest Authentication

HTTP Digest Authentication

Quick Overview

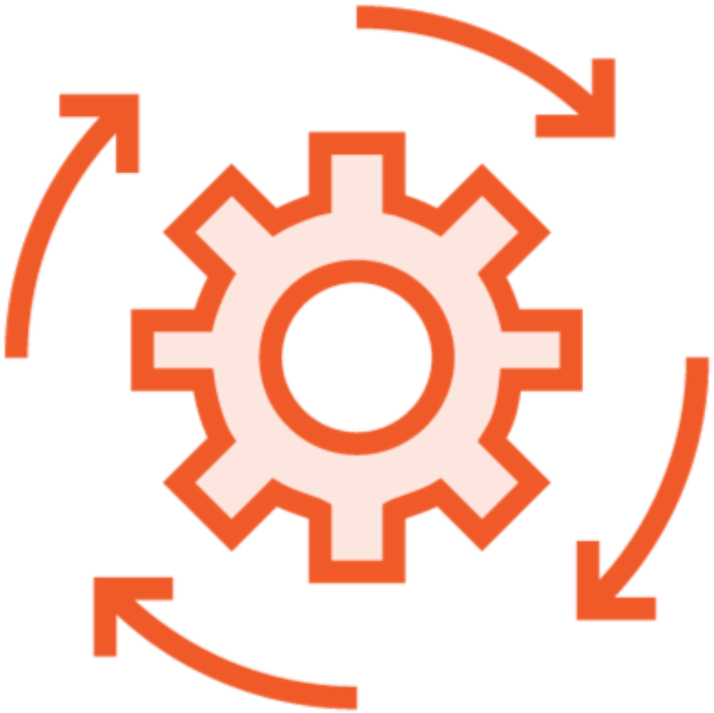
- What is HTTP Digest authentication?
- Advantages and disadvantages
- How to implement it in PHP

HTTP Digest Authentication



- Defined in RFC 2069
- Added security enhancements in RFC 2617
- More complex than Basic Authentication
- Credentials are hashed
- Uses additional security features

HTTP Digest Authentication

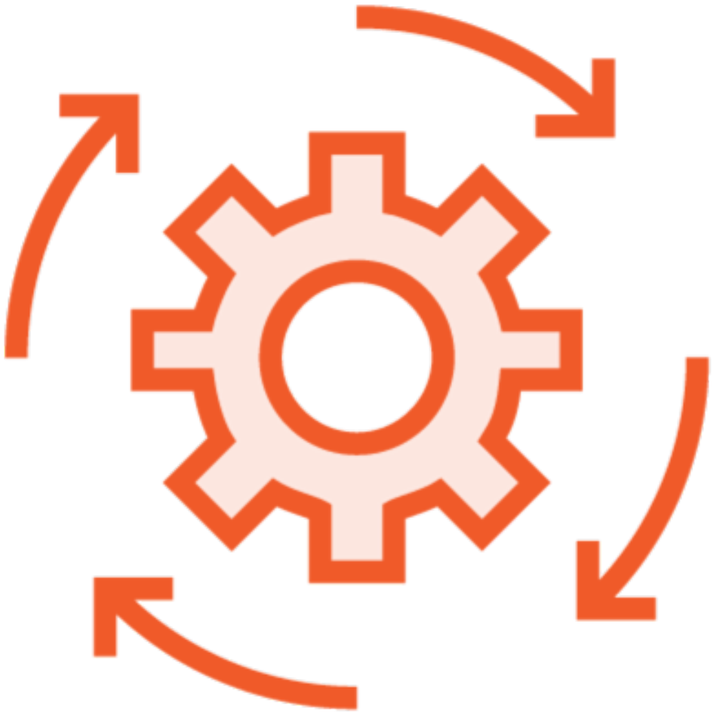


- Replacement for Basic Authentication
- Client credentials are not sent in plain text
- Uses a server-generated nonce

“A nonce is an arbitrary number that can be used just once in a cryptographic communication.”

Wikipedia - Cryptographic Nonce

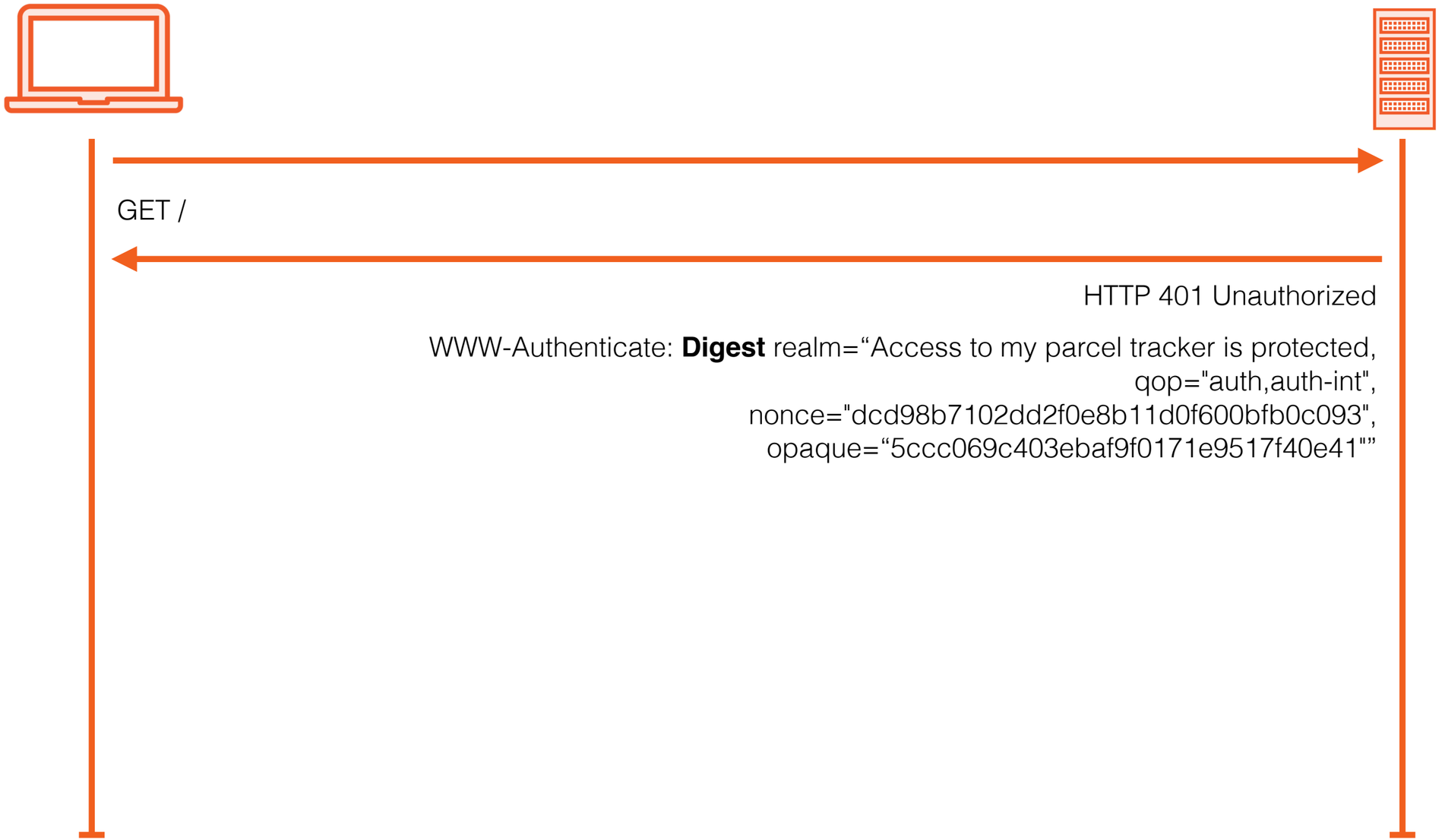
HTTP Digest Authentication



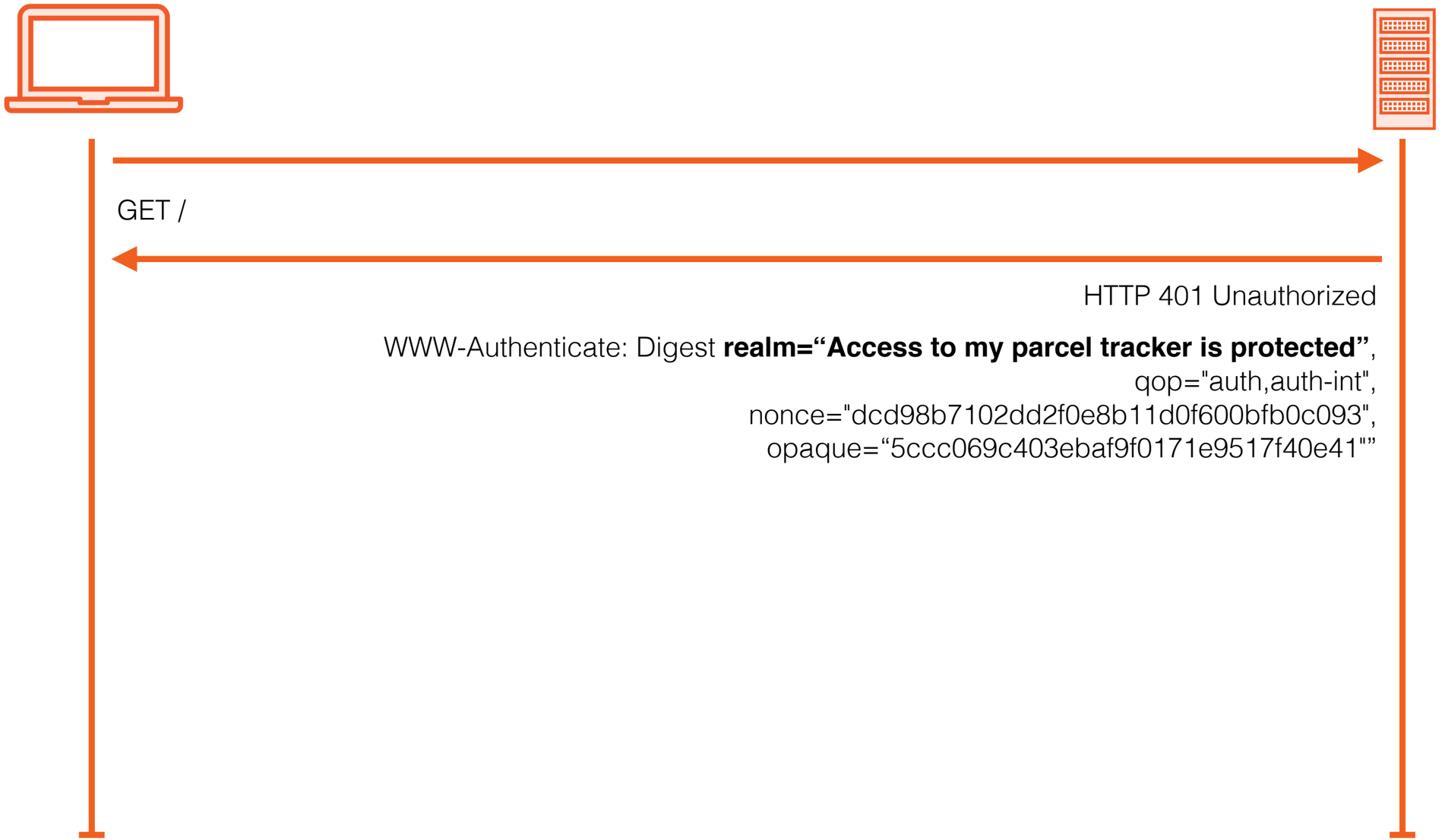
- Replacement for Basic Authentication
- Client credentials are not sent in plain text
- Uses a server-generated nonce
- Body is sent in plain text

HTTP Digest Authentication Overview

HTTP Digest Authentication Overview



HTTP Digest Authentication Overview



Additional Digest Authentication Properties

domain	An optional, quoted, space-separated list of URIs which are protected by this authentication request
opaque	A base64-encoded or hexadecimal string generated by the server and used in communication with the server.
stale	A flag that indicates that the previous request from the client was rejected because the nonce value was stale.
algorithm	This indicates the algorithm used to produce the digest. It is assumed to be MD5 if it is not specified.
nonce	A unique code generated by the server every time a 401 response is sent. It is either a base64-encoded or hexadecimal value. This helps the server guard against replay attacks.

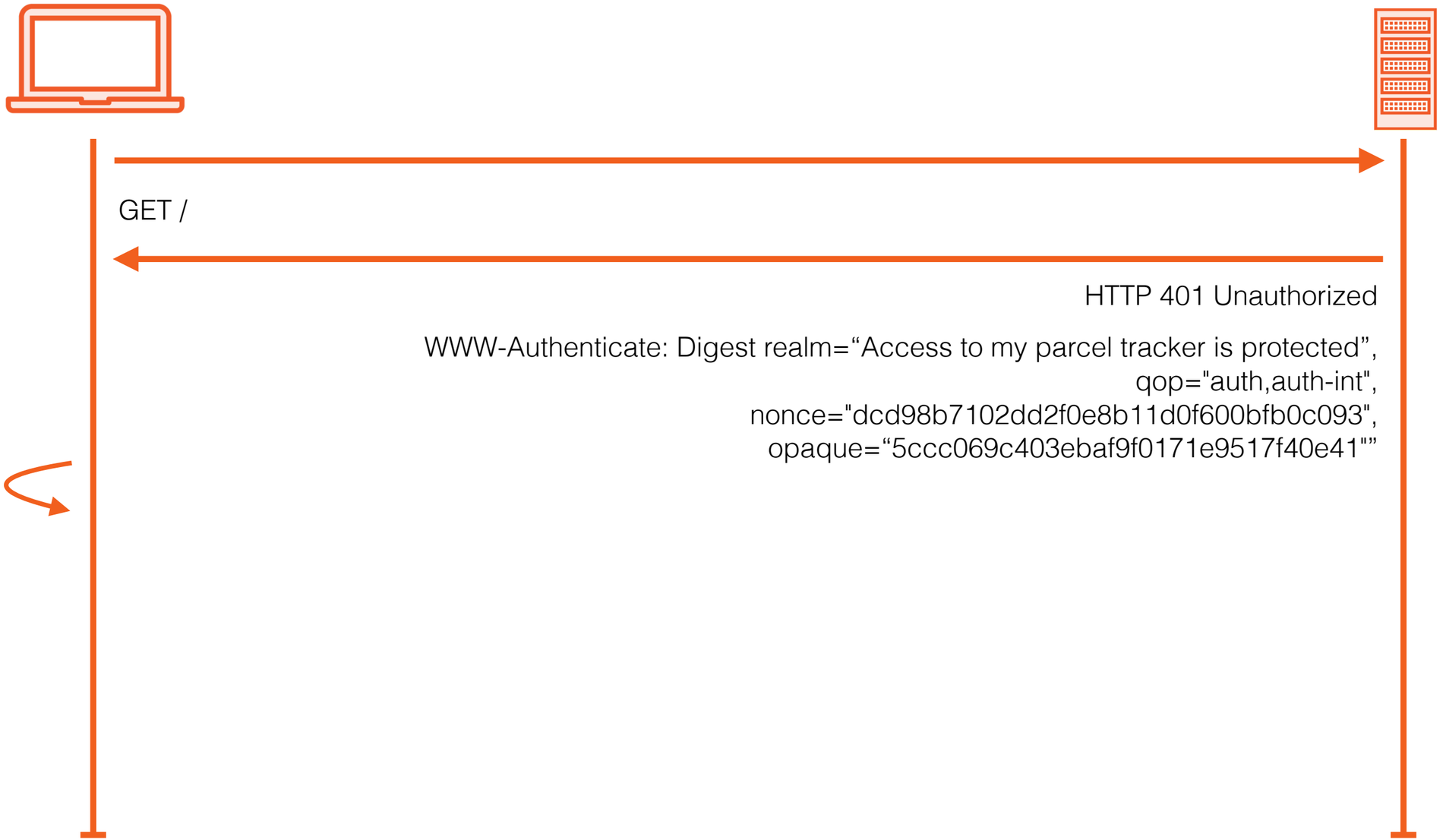
“A replay attack is a form of network attack in which a valid data transmission is maliciously or fraudulently repeated or delayed. This is carried out either by the originator or by an adversary who intercepts the data and re-transmits it, possibly as part of a masquerade attack by IP packet substitution.”

Wikipedia - Replay Attacks

Additional Digest Authentication Properties

qop	Quality of Protection. It can be set to one of `auth` or `auth-int`. This influences how the hash is created and is an integrity code for the request. If it is set to 'auth', only the requested URI will be taken into consideration. If it is 'auth-int' the body of the request will also be used in the hash.
cnonce	A unique id generated by the client. This value helps both the client and server prove that they have a known shared secret. It's required when the server sends a “qop". It must not be sent if the server did not send a qop directive.
nonce-count	Nonce count. This is a hexadecimal count of the number of requests that the client has sent with the nonce value used in the request. It allows the server to detect request replays.

HTTP Digest Authentication Overview



Generate the Authorization Header

```
HA1 = An MD5 hash of the username, password, and the realm string
HA2 = An MD5 hash of the authentication method and request URI
Response = MD5 hash of HA1, HA2, nonce, nonce-count, cnonce, and qop
```

HTTP Digest Authentication Overview

Advantages and Disadvantages

Advantages of Digest Authentication

- Part of the HTTP specification
- Password is never sent in the clear
- Includes a server nonce
- Includes a client nonce

Disadvantages of Digest Authentication

- Some header fields are optional
- Security level cannot be guaranteed
- Vulnerable to Man-in-the-middle attacks
- Prevents stronger password hashing

Quick Recap

- What is HTTP Digest authentication?
- Advantages and disadvantages
- Implemented it in PHP

Module Recap

- Learned about:
  - HTTP authentication
  - Basic and Digest authentication
  - HTTP headers
  - Why HTTPS/TLS is essential
- Implemented Basic and Digest authentication in PHP

Coming Up Next

- Form-based authentication