

Authentication



Maximilian Schwarzmüller

AUTHOR

@maxedapps www.mschwarzmueller.com



Overview



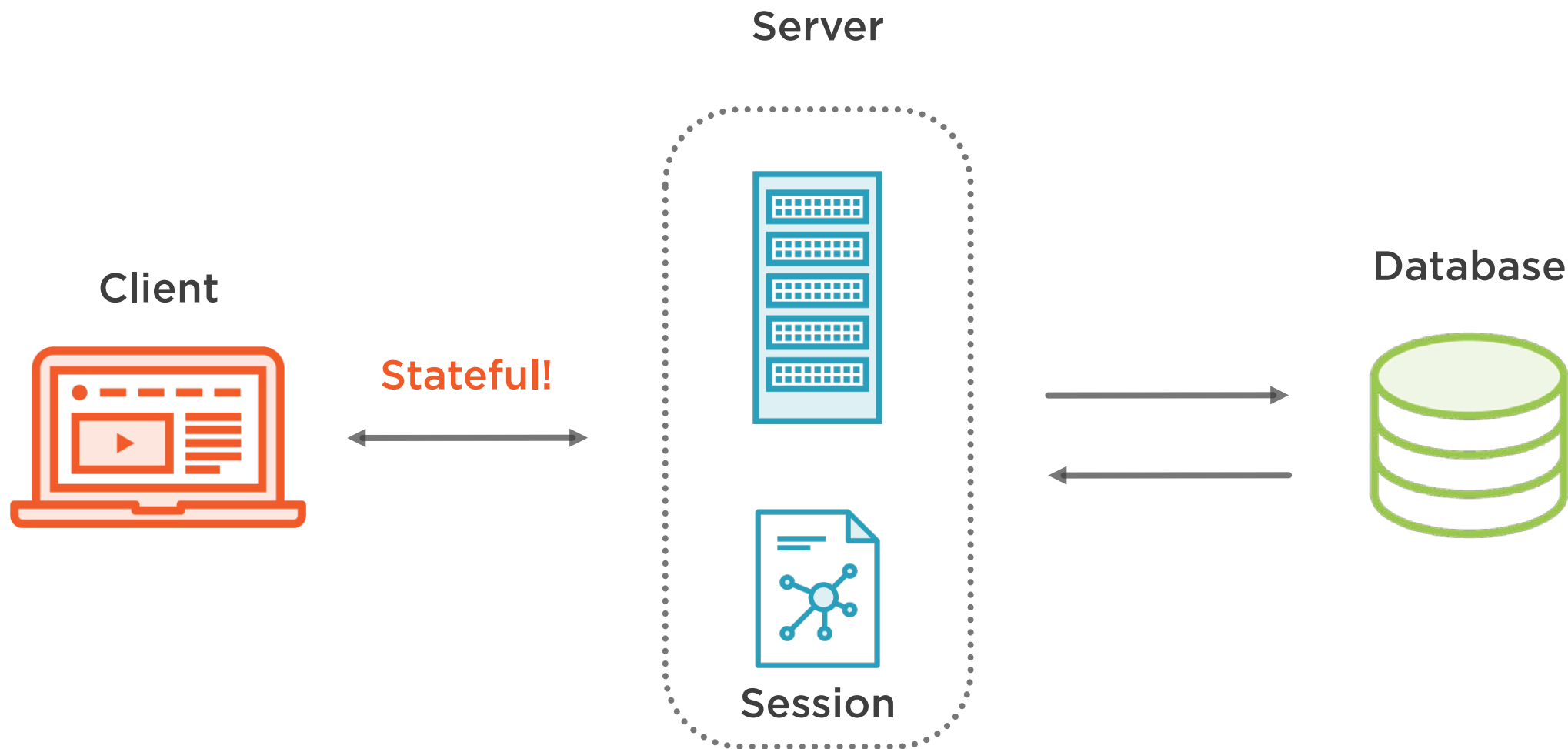
Learn why session-powered authentication doesn't work

Explore alternative approaches

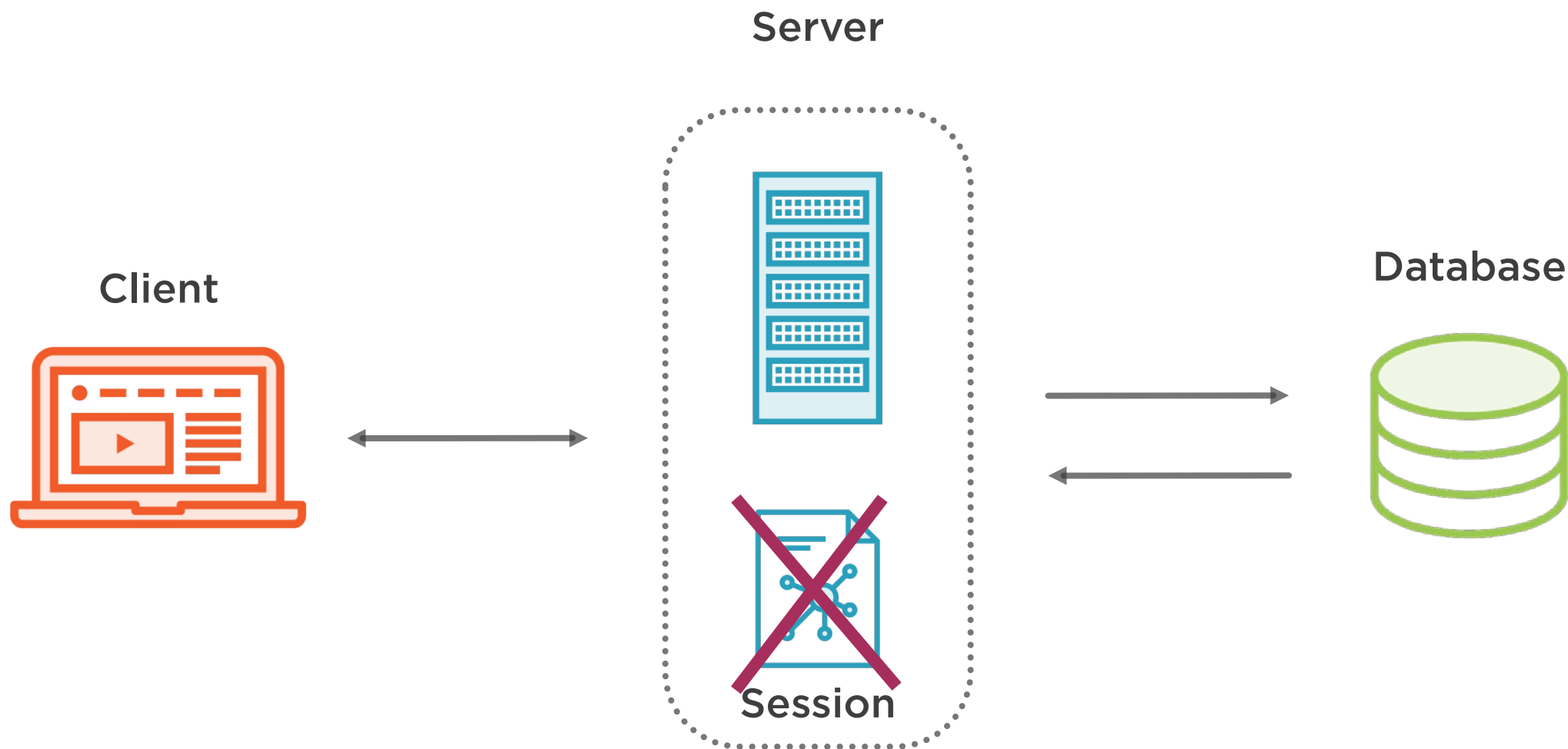
Understand how to implement JWT authentication



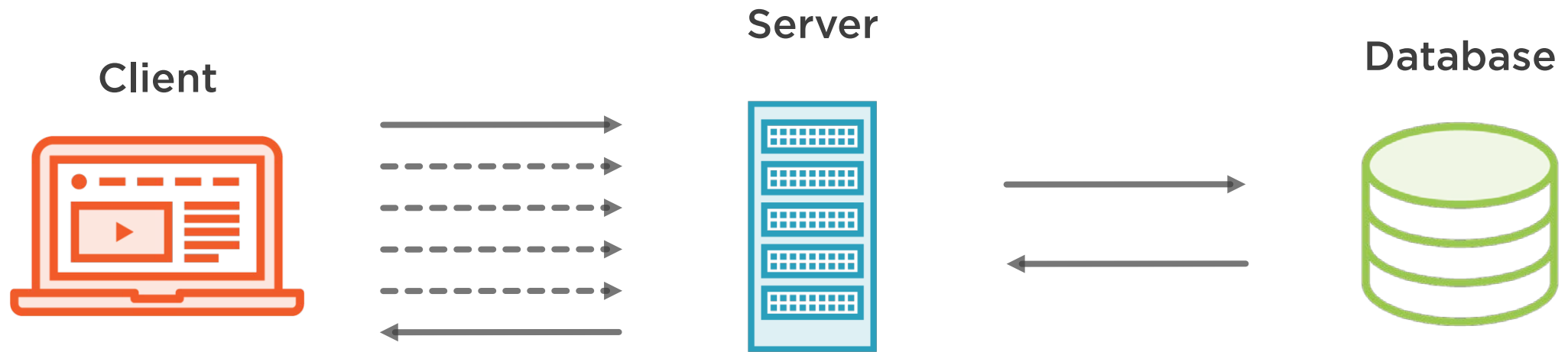
“Traditional” Authentication



RESTful Services Are Stateless!



Authenticate with Each Request

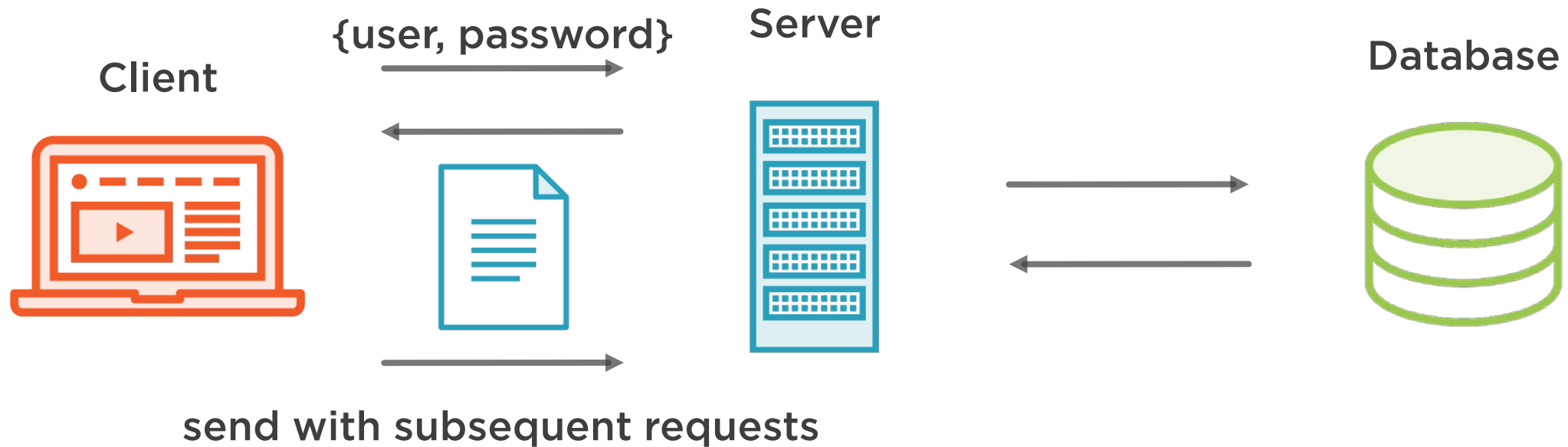


Everyone loves logging in
with each request.

NOT!



Solution: Single Sign-on



JSON Web Token (JWT) and OAuth2

JWT

Authentication Protocol

Allows encoded claims to be transferred between two parties

Token is issued upon identification

Client stores token and sends it with subsequent requests

OAuth2

Authentication Framework

Uses roles, client types & profiles, auth grants and endpoints to manage authentication

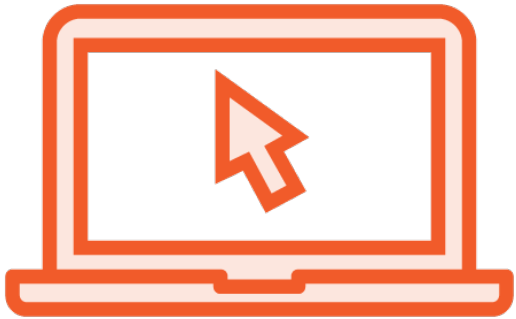
May issue JWT as auth mechanism

Error-free implementation is a complex task

Anyways: Use SSL!

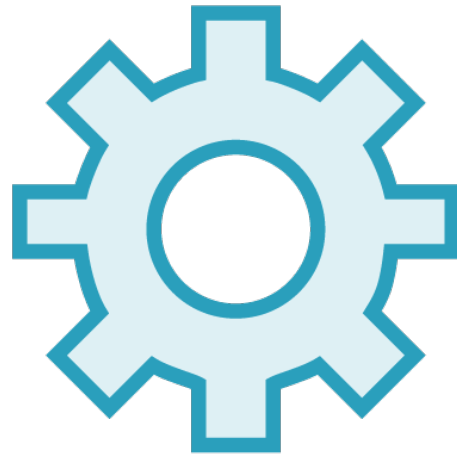


Implementing JWT in Laravel App



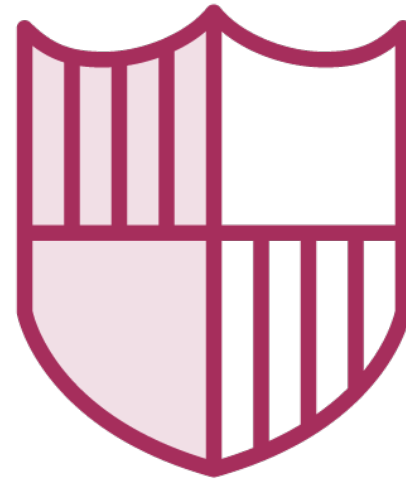
Install

Install third-party package



Configure

Set up JWT package and rules



Protect

Implement middleware to protect routes



Parse User

Retrieve authenticated user if required



Install 3rd Party Package (JWT)

```
// Edit composer.json file
```

```
"require": { "tymon/jwt-auth": "0.5.*" }
```

```
// In app/config.php, add following provider...
```

```
Tymon\JWTAuth\Providers\JWTAuthServiceProvider::class
```

```
// ... and facade
```

```
'JWTAuth' => Tymon\JWTAuth\Facades\JWTAuth::class
```

```
// Publish the package config file
```

```
php artisan vendor:publish
```



Configure JWT Package

```
// Generate random key to sign tokens
```

```
php artisan jwt:generate
```

```
// In config/jwt.php, configure token
```

```
// Example: Time to live
```

```
'ttl' => 60
```

```
// Example: Underlying User Model
```

```
'user' => 'App\User'
```



Sign In with JWT

```
try {  
    if (! $token = JWTAuth::attempt($credentials)) {  
        return response()->json(['error' => '...'], 401);  
    }  
} catch (JWTException $e) {  
    return response()->json(['error' => '...'], 500);  
}
```



Authenticate Requests with JWT

```
$this->middleware('jwt.auth');
```

```
$user = JWTAuth::parseToken()->authenticate()
```

