

# Web-Development Backend

JOI und JWT - Server vor invalidem Input schützen  
oder:

Wie mache ich den Server kaputt?

[webdev.andreasnicklaus.de](https://webdev.andreasnicklaus.de)

# Ungeschützte Architektur



# Geschützte Datenbankstruktur



# Beispiel Twitter

Welche Arten von schlechten Inputs können wir erfahren?

```
{  
  "authorId" : "ElonMusksUserId",  
  "content" : "Mark Zuckerberg is a great business man.",  
  "creationTime": "2024-01-12T18:13:38.699Z"  
}
```

# Beispiel Twitter

## Schlechte Inputs

```
{  
  "authorId" : "iReallyAmElonMusk",  
  "content" : "Mark Zuckerberg $$%&/&",  
  "creationTime": "2024-01-12T18:13:38.699Z"  
}
```

```
{  
  "content" : 123456,  
  "creationTime": "1603-01-01T00:00:00.000Z"  
}
```

# SQL Injection

```
{  
  "authorId" : "ElonMuskUserId"; DELETE * from Users; COMMIT;",  
  "content" : "123456"  
}
```

```
SELECT * from Users where id='ElonMuskUserId'; DELETE * from Users; COMMIT;
```

# Schlechte Inputs

- Felder nicht gefüllt
- Felder nicht vorhanden
- Felder haben nicht den richtigen Typ
- Inhalt ist nicht erlaubt

# Inputvalidierung mit Joi

```
const myTwitterPostSchema = joi.object({  
  authorId: joi.string().alphanum().required(),  
  content: joi.string()  
})
```