Challenges in Continuous Security Compliance

Florian Angermeir
20.11.2024

fortiss

BLEKINGE TEKNISKA HÖGSKOLA · BTH ·

About Me

Currently

- Researcher at fortiss
- PhD student at Blekinge Institute of Technology

Past

- Research and implementation of security compliance in agile/DevOps project for 5 years in industry
- System administrator at Technical University of Munich for 5 years

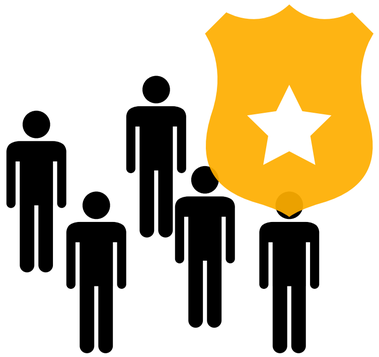
Why Security?

Why Security?

Secure Product

Why Security?

Secure Product

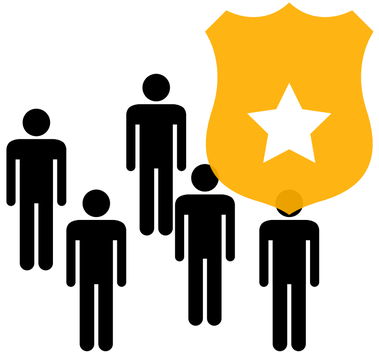


Protect Customers

Why Security?

Secure Product

Protect Business or Reputation

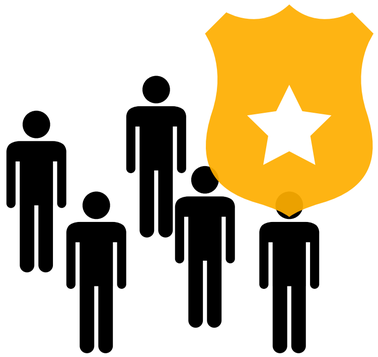


Protect Customers

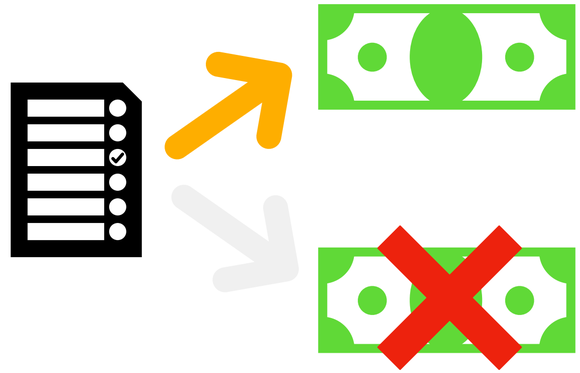
Why Security?

Secure Product

Protect Business or Reputation



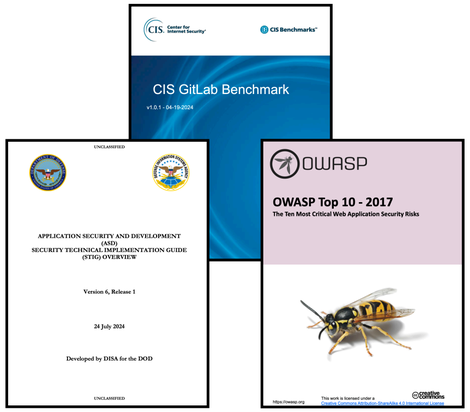
Protect Customers



Requirement to Enter Market

Available Guidance For Security

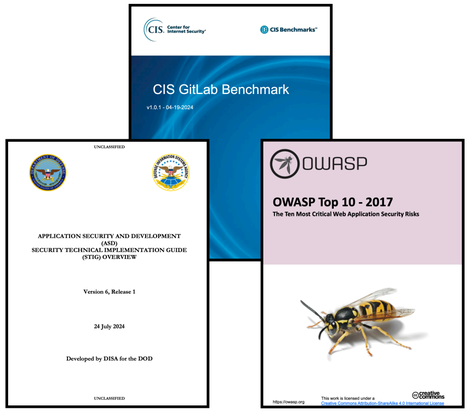
Available Guidance For Security



Best Practices

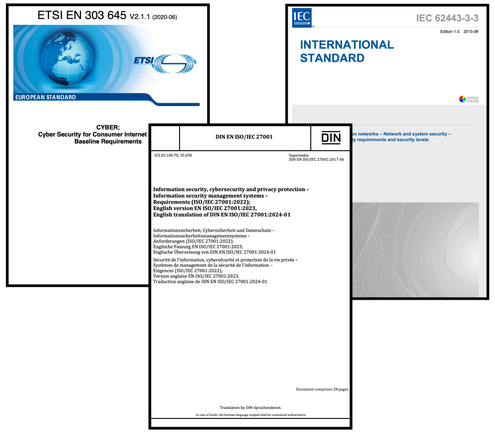
- Voluntary guidelines
- Cover the most common security issues
- Often easily applicable

Available Guidance For Security



Best Practices

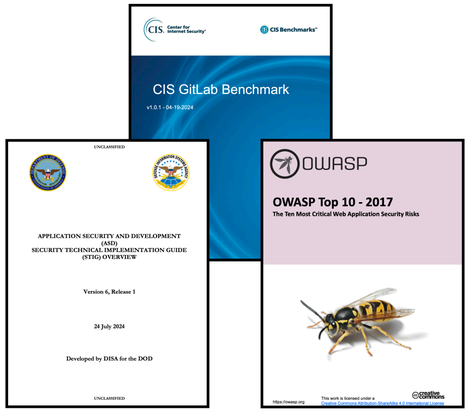
- Voluntary guidelines
- Cover the most common security issues
- Often easily applicable



Security Standards

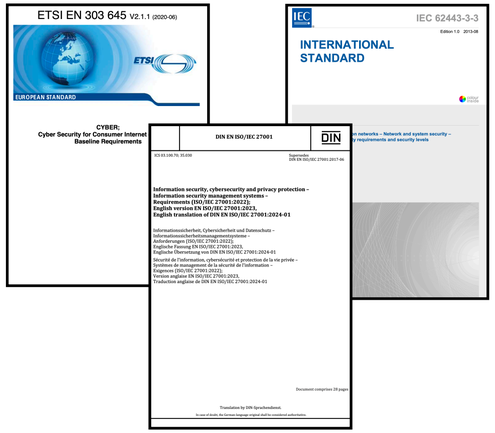
- Voluntary/Mandatory guidelines
- Offer high level of security posture
- Provided by standardisation bodies (e.g. ISO)

Available Guidance For Security



Best Practices

- Voluntary guidelines
- Cover the most common security issues
- Often easily applicable



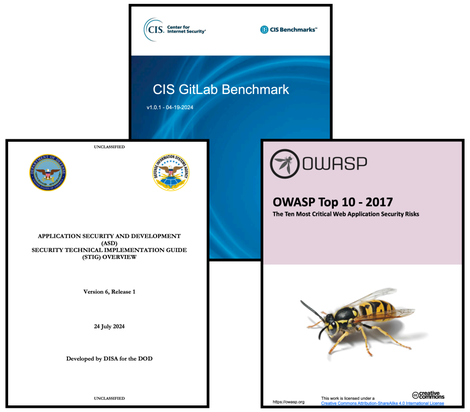
Security Standards

- Voluntary/Mandatory guidelines
- Offer high level of security posture
- Provided by standardisation bodies (e.g. ISO)

Regulations

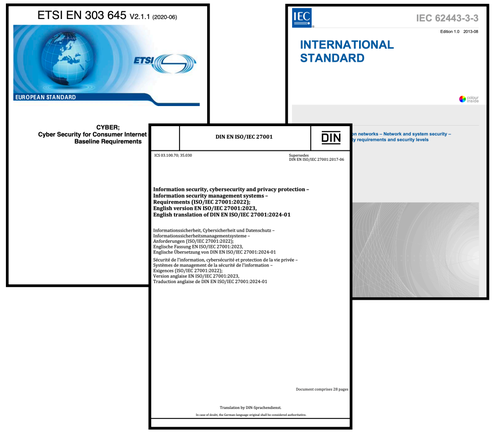
- Mandatory guidelines
- Dictate security posture - explicit consequences
- Provided by governmental authorities (e.g. EU)

Available Guidance For Security



Best Practices

- Voluntary guidelines
- Cover the most common security issues
- Often easily applicable



Security Standards

- Voluntary/Mandatory guidelines
- Offer high level of security posture
- Provided by standardisation bodies (e.g. ISO)

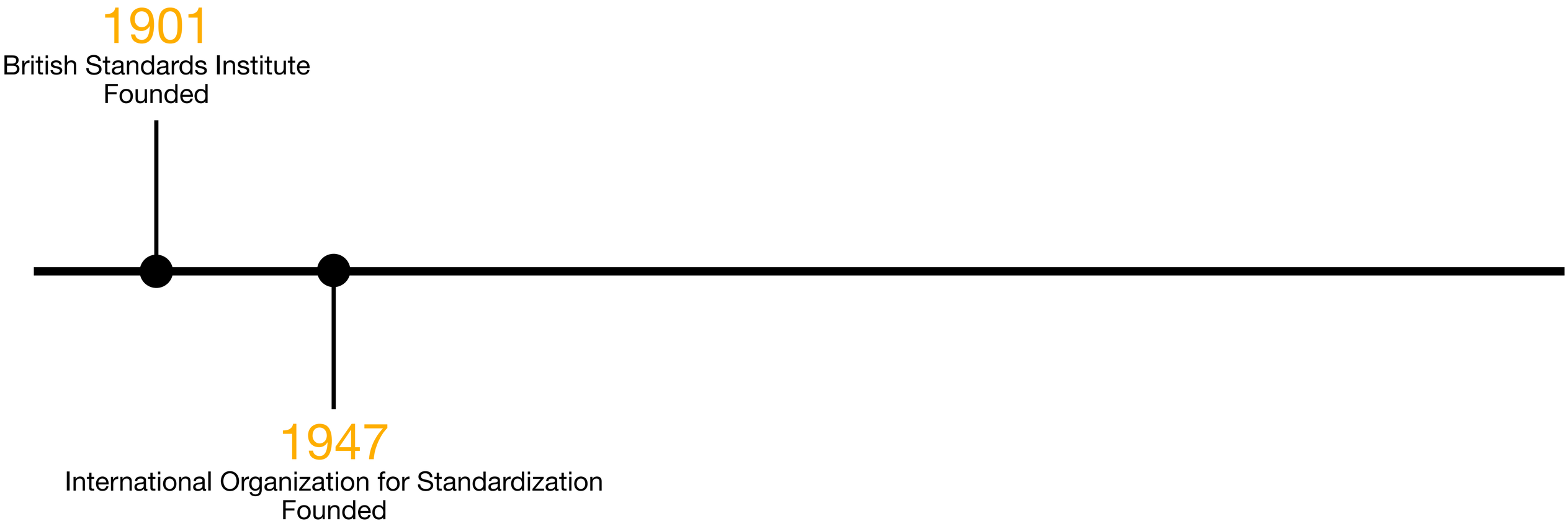
Regulations

- Mandatory guidelines
- Dictate security posture - explicit consequences
- Provided by governmental authorities (e.g. EU)

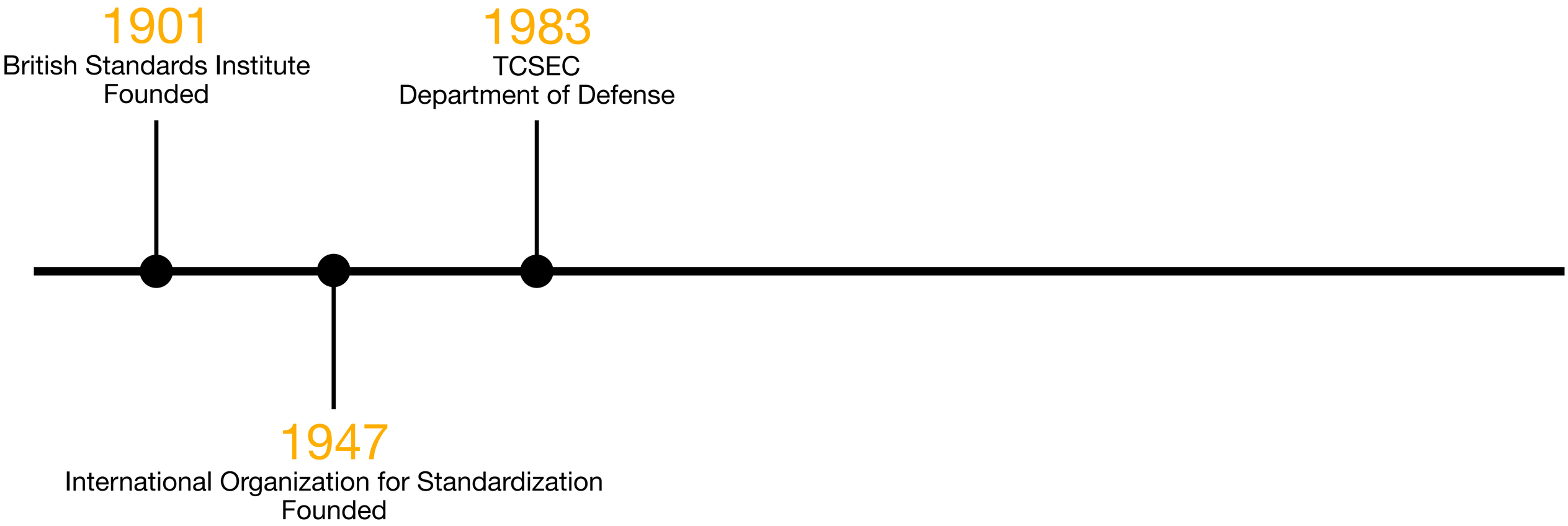
Security Standards

Brief History of Security Standards

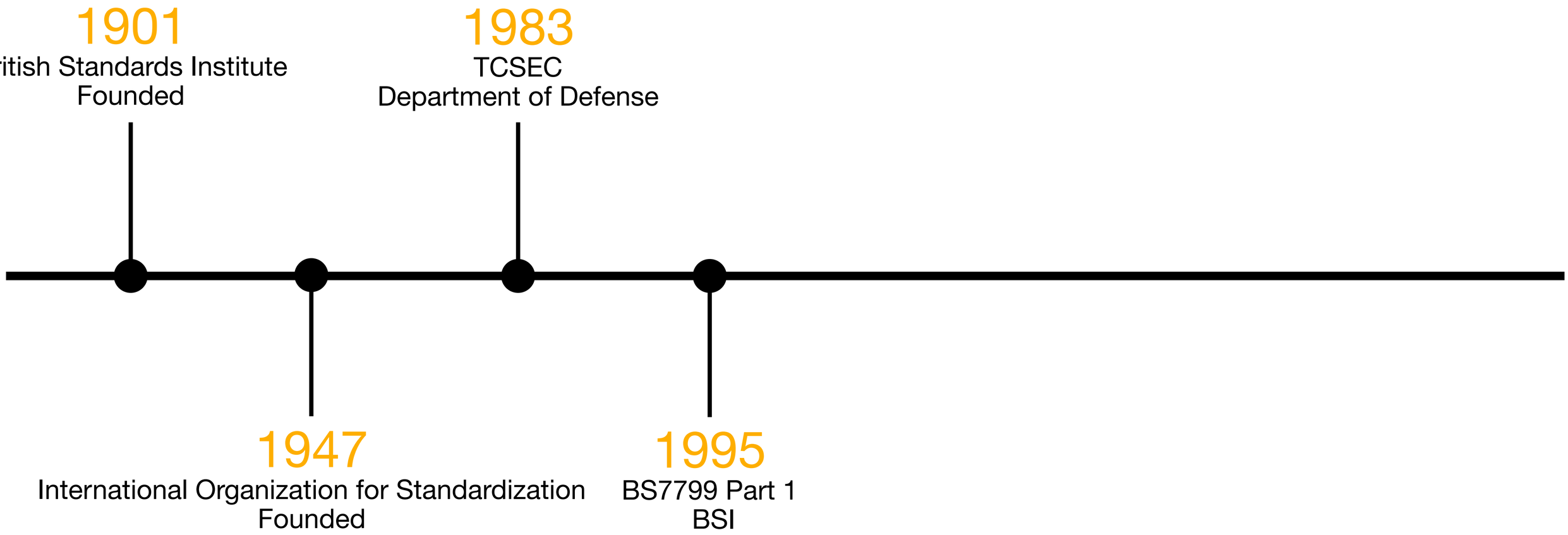
Brief History of Security Standards



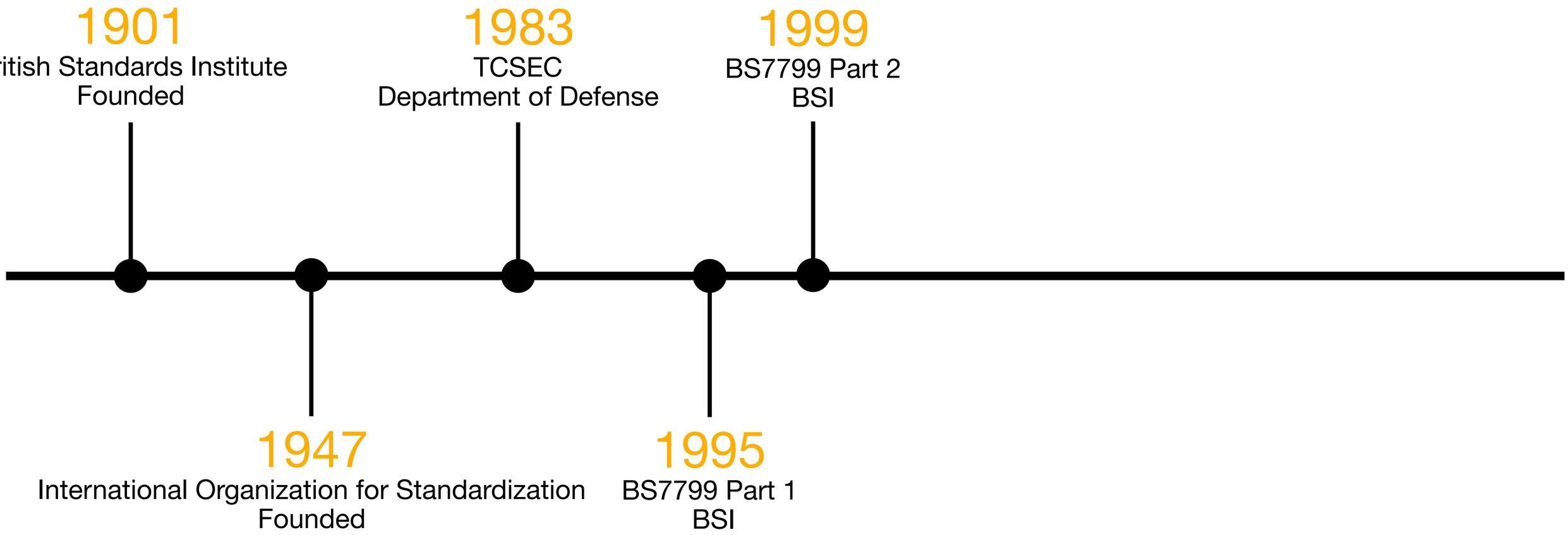
Brief History of Security Standards



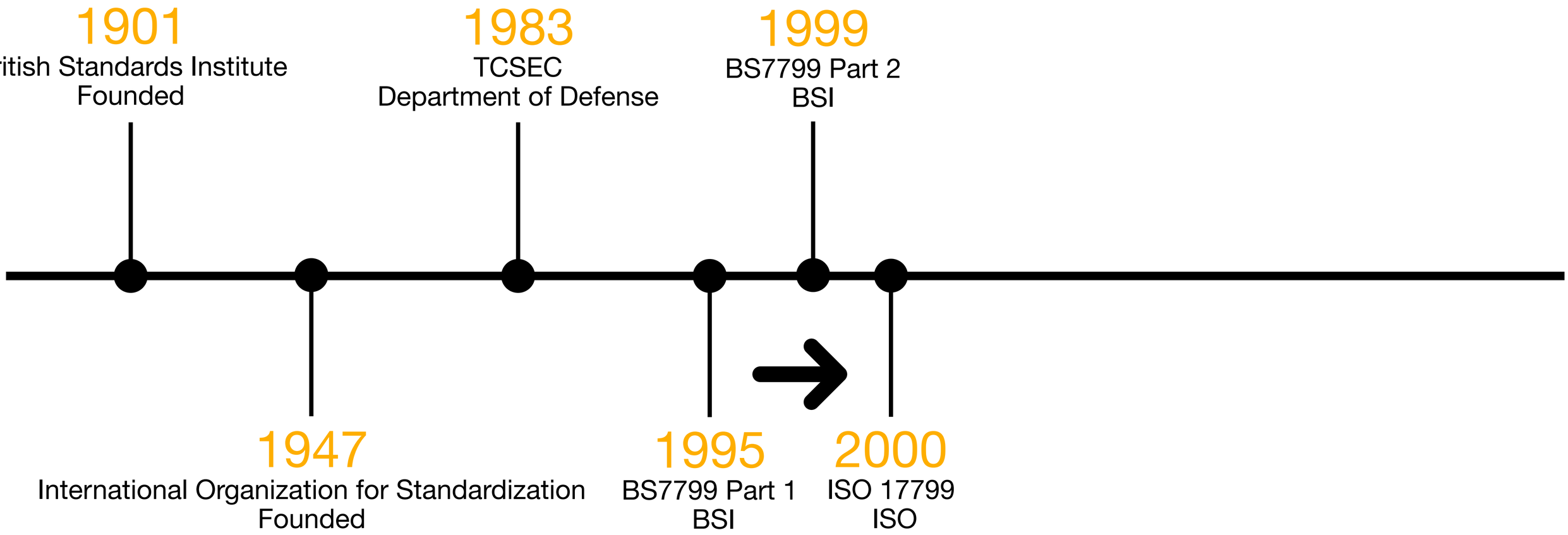
Brief History of Security Standards



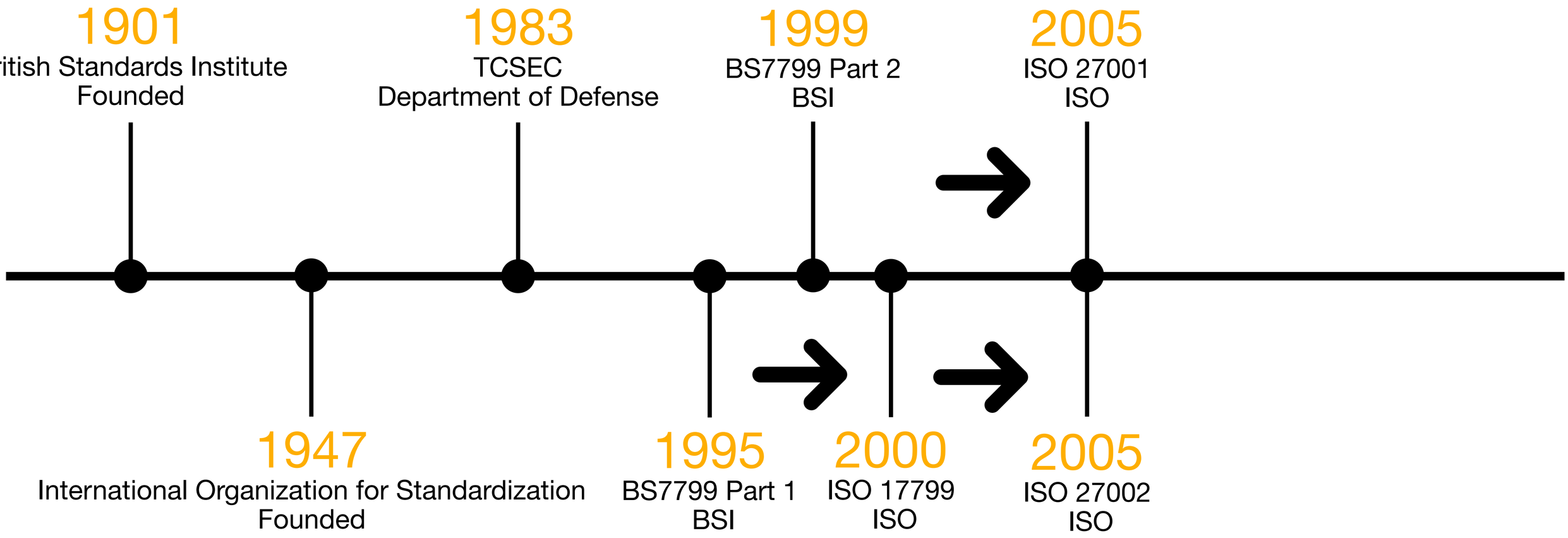
Brief History of Security Standards



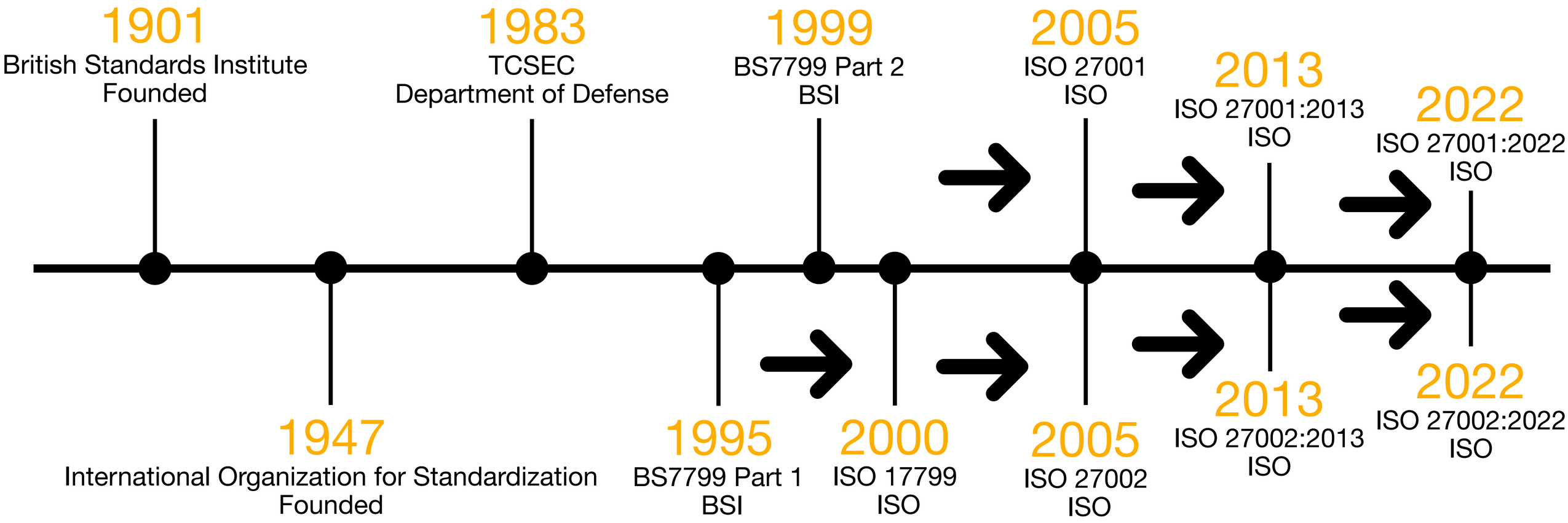
Brief History of Security Standards



Brief History of Security Standards



Brief History of Two Security Standards



Ever Present Challenges

Ever Present Challenges

1. Security standards are difficult to understand & Interpretative

Ever Present Challenges

1. Security standards are difficult to understand & Interpretative
2. Stakeholders lack security knowledge

   Security is a big field - you can't know everything, especially if is not your job

Ever Present Challenges

1. Security standards are difficult to understand & Interpretative
2. Stakeholders lack security knowledge

   Security is a big field - you can't know everything, especially if is not your job

3. Security experts scarce

   1 (=One) Security expert for 100 developers [1]

Ever Present Challenges

1. Security standards are difficult to understand & Interpretative
2. Stakeholders lack security knowledge

   Security is a big field - you can't know everything, especially if is not your job

3. Security experts scarce

   1 (=One) Security expert for 100 developers [1]

4. Ensuring you satisfy security requirements (e.g. audits) is resource-intensive

[1] https://www.sonatype.com/hubfs/SON_Survey2018_final.pdf

Contemporary Product Development

DevOps, Agile & Security Compliance

Modern Software Development

Short release cycles
Testing continuously
Change is the only constant
"Working code over documentation"

DevOps, Agile & Security Compliance

Modern Software Development

Short release cycles
Testing continuously
Change is the only constant
"Working code over documentation"

Traditional Security Compliance

Resource-intensive
Process oriented
Doesn't handle frequent changes
Requires extensive documentation

DevOps, Agile & Security Compliance

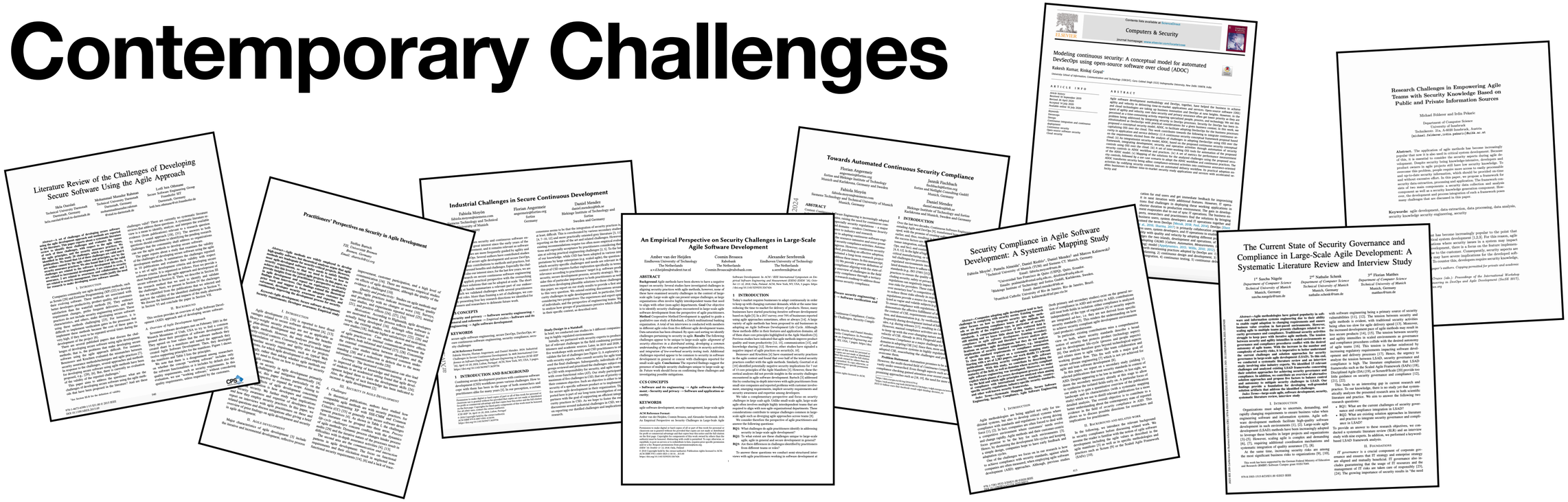
Modern Software Development

Short release cycles
Testing continuously
Change is the only constant
"Working code over documentation"

Traditional Security Compliance

Resource-intensive
Process oriented
Doesn't handle frequent changes
Requires extensive documentation

Contemporary Challenges



- Workshop with 3 companies over 2 years (Moyón et al. [1])
- Literature review (Angermeir et al. [2]) & Survey
- 27 challenges in 6 categories

[1] https://arxiv.org/pdf/2401.06529
[2] https://arxiv.org/pdf/2407.21494

Contemporary Challenges

Category: Security in Continuous Development
e.g. Perform threat modelling and consistently increment/adapt it throughout sprints

Category: Security in the Value Stream
e.g. Prioritisation of security requirements vs. system functionalities

Category: Security Implementation Efficiency
e.g. Security compliance evidence generation and documentation too time consuming

Category: Security Knowledge
e.g. Enable security knowledge and ownership in engineering teams

Category: Security into CI/CD pipelines
e.g. Achieve efficient handling of security tool findings and involve into regular issue handling process

Category: Security Strategy Success
e.g. Insufficient leadership on security

Source: https://arxiv.org/pdf/2407.21494

Let's Make One Deeper Dive

Challenge Gist
Ensuring security requirements satisfaction over the entire software development life-cycle is difficult in fast-paced development.

Let's Make One Deeper Dive

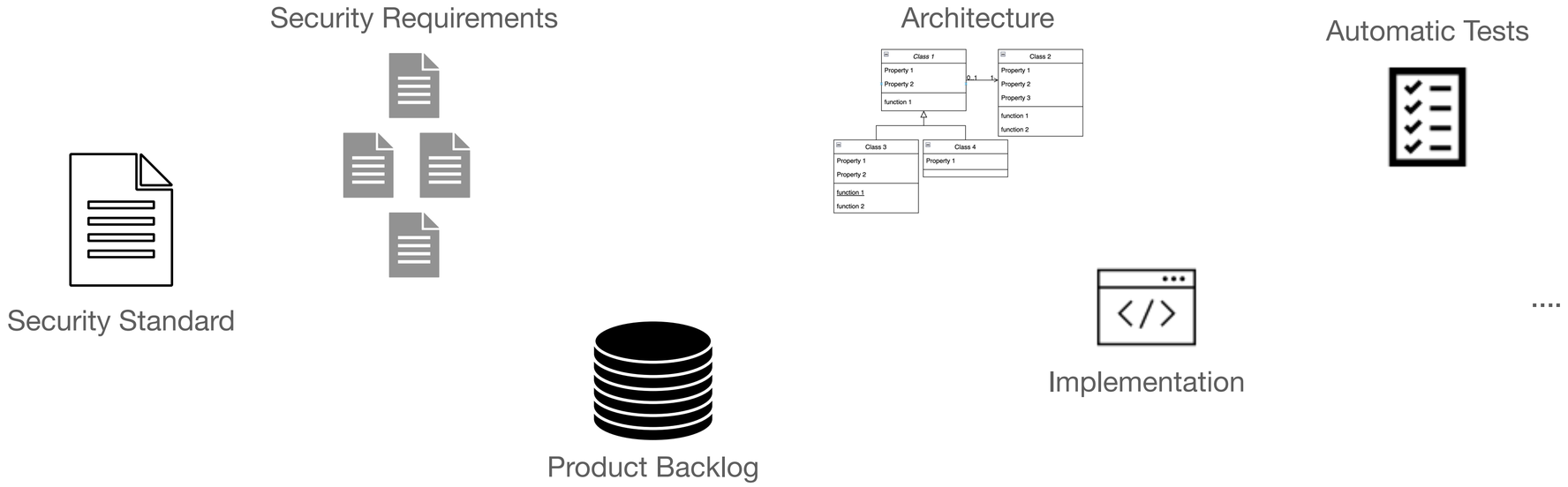
Challenge Gist
Ensuring security requirements satisfaction over the entire software development life-cycle is difficult in fast-paced development.

Solution (Continuous automatic security compliance)
Automatic artefact generation & traceability over the entire workflow

Let's Make One Deeper Dive

Challenge Gist
Ensuring security requirements satisfaction over the entire software development life-cycle is difficult in fast-paced development.

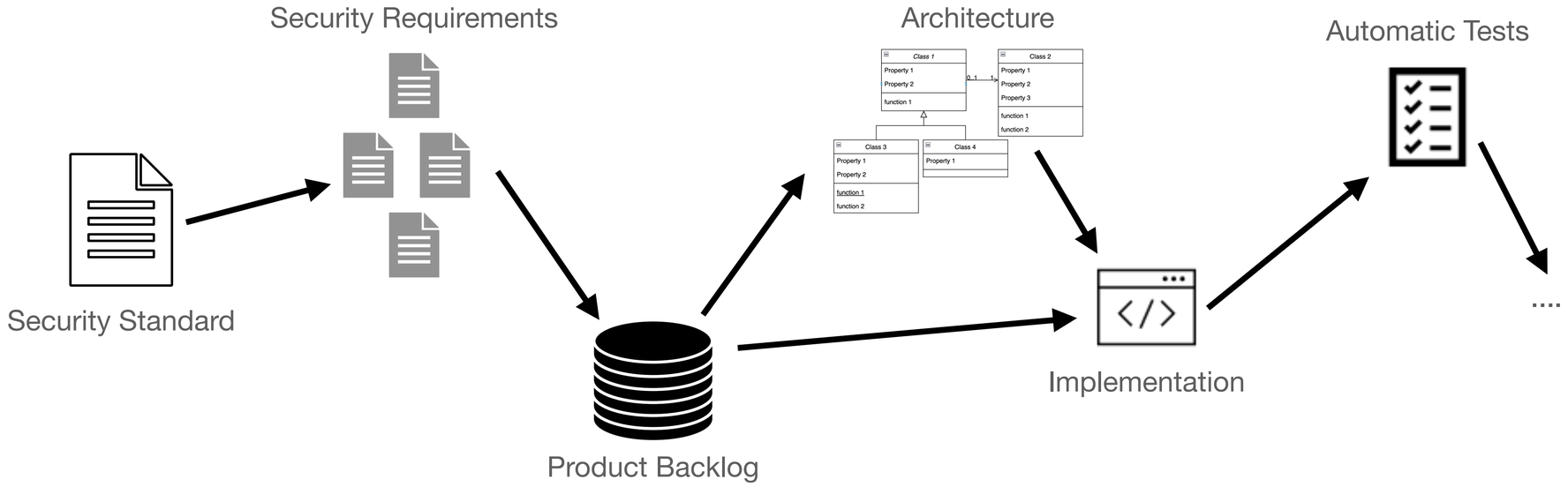
Solution (Continuous automatic security compliance)
Automatic artefact generation & traceability over the entire workflow



Let's Make One Deeper Dive

Challenge Gist
Ensuring security requirements satisfaction over the entire software development life-cycle is difficult in fast-paced development.

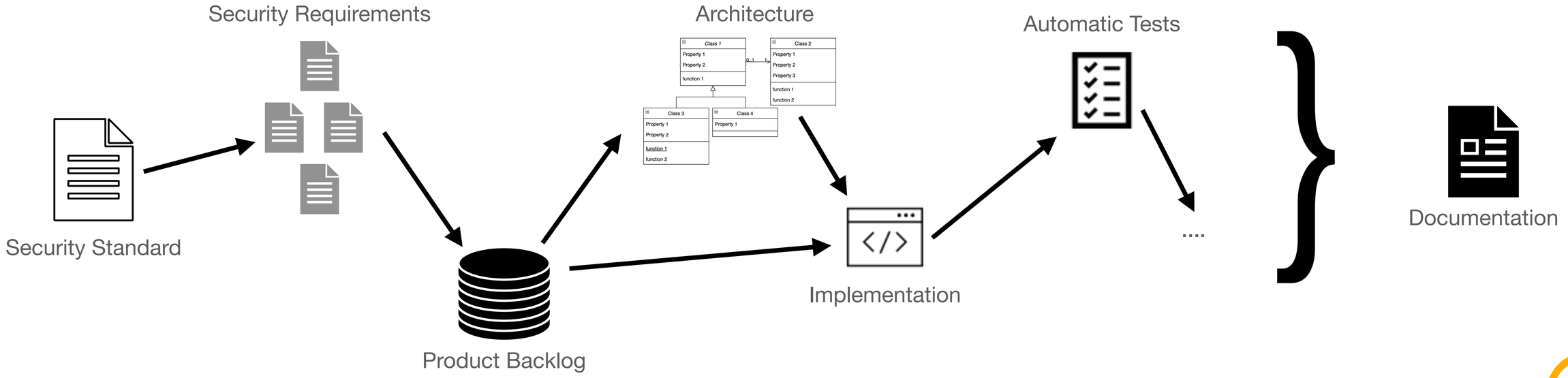
Solution (Continuous automatic security compliance)
Automatic artefact generation & traceability over the entire workflow



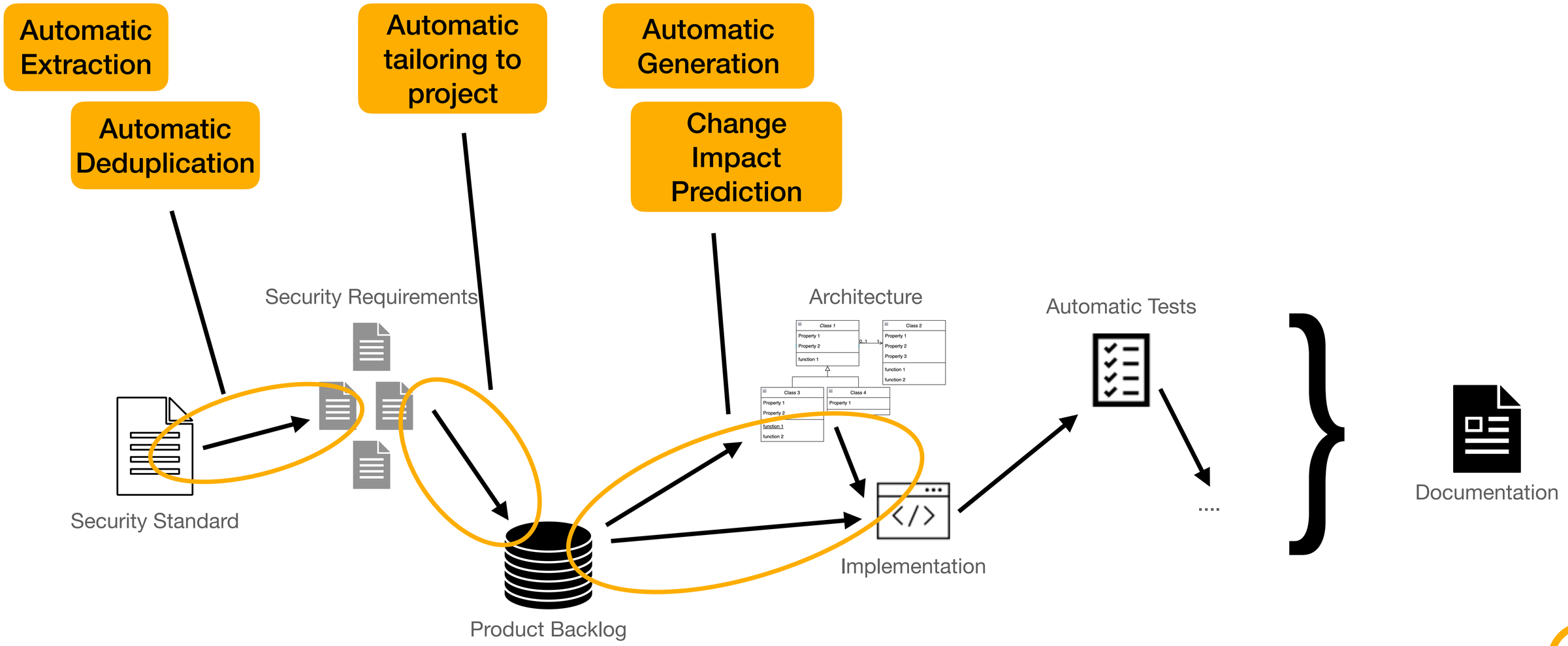
Let's Make One Deeper Dive

Challenge Gist
Ensuring security requirements satisfaction over the entire software development life-cycle is difficult in fast-paced development.

Solution (Continuous automatic security compliance)
Automatic artefact generation & traceability over the entire workflow



Let's Make One Deeper Dive



Final Consideration

Security as End in Itself?

Often security - especially if imposed from outside - feels like an end in itself.

Security as End in Itself?

Often security - especially if imposed from outside - feels like an end in itself.

BUT...

Most often its not - it serves a purpose for our business (goals or constraints).

Security as End in Itself?

Often security - especially if imposed from outside - feels like an end in itself.

BUT...

Most often its not - it serves a purpose for our business (goals or constraints).

NEVERTHELESS...

Too often business goals and constraints conflict.

Security as End in Itself?

Often security - especially if imposed from outside - feels like an end in itself.

BUT...

Most often its not - it serves a purpose for our business (goals or constraints).

NEVERTHELESS...

Too often business goals and constraints conflict.

My Takeaway

Security needs to become better at supporting other business goals.

Let's Discuss!

Takeaway 1
We need new ways to ensure security compliance in agile and DevOps

Takeaway 2
Security has to be integrated with minimal impact on other business goals

florian.angermeir@bth.se
https://angermeir.me

Contact

Florian Angermeir
florian.angermeir@bth.se
https://angermeir.me

fortiss GmbH, Germany
Blekinge Institute of Technology, Sweden

Structure to Approach Security

Organisational Security

- Is information classified according to business needs? (ISO 27001 5.12)
- Is information security addressed in supplier agreements? (ISO 27001 5.20)

Product Development Process Security

- Is a threat modelling analysis process in place? (IEC 62443-4-1 SR 2)
- Does the product design document external interfaces? (IEC 62443-4-1 SD 1)

Product Security

- Does the product limit unsuccessful login attempts? (IEC 62443-3-3 SR 1.11)
- Is the software application running with least necessary privileges? (ETSI EN 303 645 5.6-7)

Most Pressing Challenges

Challenge 1: Prioritisation of Security Requirements vs. System Functionalities

Challenge 2: Make Security Architecture visible in Backlog and Documentation

Challenge 3: Include Security Activities into Continuous Deployment

Most Pressing Challenges

Challenge 1: Prioritisation of Security Requirements vs. System Functionalities

Challenge 2: Make Security Architecture visible in Backlog and Documentation

Challenge 3: Include Security Activities into Continuous Deployment

---

Solution 1: Agile/DevOps team security skill improvement	Challenge 1
Solution 2: Implementation of continuous security feedback loop	
Solution 3: Continuous automatic security compliance	Challenge 2,3
Solution 4: Visibility & assessment of security practices maturity	Challenge 2

Automation, the Saviour?

Automating everything sounds tempting -> saves resources?

BUT

Is it theoretically possible to automate everything?

No! For example only 31% of the IEC 62443-4-1 can be full automated [1]

Should we automate everything possible?

Is it feasible? Is the outcome better? Does it actually save resources?

My Takeaway

Sensible Integration: Automation where necessary and impactful.

[1] https://arxiv.org/pdf/2105.13024