From Samples to Data Streams

Auditing organisations that change a thousand times a day

Germán Fuentes Capella

Germán sounds like "Herman"

CTO, trigosec

german@trigosec.com

A quick show of hands

Do you audit environments where:

1. **Thousands of changes occur during one audit period?**

german@trigosec.com

A quick show of hands

Do you audit environments where:

1 Thousands of changes occur during one audit period?

2 **The implementation of SDLC controls changes during the audit period?**

german@trigosec.com

A quick show of hands

Do you audit environments where:

1. Thousands of changes occur during one audit period?
2. The implementation of SDLC controls changes during the audit period?
3. **Different teams use different pipelines or release processes?**

A quick show of hands

Do you audit environments where:

1. Thousands of changes occur during one audit period?
2. The implementation of SDLC controls changes during the audit period?
3. Different teams use different pipelines or release processes?
4. **Evidence is spread across systems and takes significant effort to connect?**

You have just identified four signals

1. **Population scale**
   Thousands of change events

2. **Controls change over time**
   Control implementations change during the audit period

3. **Organisational variation**
   Teams implement controls differently

4. **Evidence across systems**
   Evidence must be connected across systems

Germán Fuentes Capella

german@trigosec.com

1. Population scale

How many software delivery teams does your organisation have?

1. Population scale | 2. Controls change over time | 3. Organisational variation | 4. Evidence across systems

Germán Fuentes Capella

german@trigosec.com

250,000

deployment events per year

10 teams × 100 deployments per day × 250 working days

A 50-item sample observes 0.02%

What if the sample is not enough to draw a reliable conclusion?

german@trigosec.com

Ways to adapt the audit approach

Response	Example
Revisit the scope	Audit one critical service rather than the organisation-wide SDLC
Redesign the sample	Increase the sample and include more emergency or high-risk changes
Stratify the population	Sample each team, pipeline, or change type separately
Analyse the full population	Test defined control conditions across every observable release

Full-population analysis shows where to look deeper

Test defined conditions across every observable event, then filter and compare the results.

Dimension	Question
Team	Is one team driving most exceptions?
Pipeline	Is one pipeline missing or bypassing controls?
Control	Which control fails most frequently?
Time	Did the problem begin after a particular change?

german@trigosec.com

Example: emergency change rate

> **Emergency production changes per 100 planned production changes**

$$\frac{\text{Emergency production changes}}{\text{Planned production changes}} \times 100$$

Increased reliance on emergency paths may increase exposure where controls are reduced or deferred.

Emergency changes rise sharply in Q4

Quarter	Emergency changes per 100 planned changes
Q1	4
Q2	6
Q3	12
Q4	**31**

german@trigosec.com

Filter Q4 by product

Product	Emergency changes per 100 planned changes
Product A	5
Product B	7
Product C	**62**

Next filter: which service within Product C explains the result?

german@trigosec.com

Filter Product C by service

Service group	Emergency changes per 100 planned changes
Existing services	8
New service	**84**

Deeper audit work finds that the new service does not have sufficient back-office support.

2. Controls change over time

A control change can create multiple control environments within one audit period

1. Population scale
2. Controls change over time
3. Organisational variation
4. Evidence across systems

Germán Fuentes Capella

german@trigosec.com

Identify what changed

Control implementation	Changes	Pipelines affected
Approval workflow	2	8
Static-analysis gate	7	12
Emergency deployment path	3	2
Production permissions	4	15

The static-analysis gate changed seven times across 12 pipelines. We select it for deeper analysis based on its reach and risk impact.

Understand expected behaviour

Before analysing events, establish what effective operation looks like:

- Where is it implemented?
- Is enforcement blocking, advisory, or manual?
- How can its configuration change?
- What evidence does it produce?
- How is a change rolled out or reversed?

The nature of the control determines which evidence can demonstrate effectiveness.

german@trigosec.com

Example: changing the static-analysis control

During the audit period, the organisation migrated its pipelines to a new static-analysis tool.

The quality gate is configured through the tool's web interface.

Expectation: a quality-gate failure caused by insufficient test coverage or identified vulnerabilities prevents deployment.

The audit must determine whether the new gate remained blocking across every pipeline affected by the rollout.

german@trigosec.com

Did the changed control remain effective?

Use analytics to compare static-analysis results with pipeline decisions across the affected population:

Static-analysis result	Deployed	Aborted	Total
Pass	920	0	920
Fail	15	65	80
Total	**935**	**65**	**1,000**

german@trigosec.com

Where did effectiveness diverge?

Pipeline	Failed analyses	Deployed after failure	Abort rate after failure
Customer API	25	0	100%
Payments	20	0	100%
Internal services	18	0	100%
Legacy platform	**17**	**15**	**12%**

The aggregate result concealed one pipeline configured in **advisory, non-blocking mode**.

german@trigosec.com

3. Organisational variation

The same control objective can be implemented differently and produce different outcomes

1. Population scale
2. Controls change over time
3. Organisational variation
4. Evidence across systems

Germán Fuentes Capella

german@trigosec.com

Map how the control is implemented

Standardised implementation

One mechanism applies across the population. Evidence is produced consistently, so one effectiveness test can be applied directly.

Variable implementation

Multiple mechanisms implement the same control objective. Evidence and results must also be assessed by implementation variant.

german@trigosec.com

Define a common effectiveness test

Across every implementation, establish:

- the control outcome that must be achieved,
- the conditions that demonstrate effective operation,
- any valid alternative paths or exceptions.

> **The mechanisms and evidence may differ, but the control outcome must remain comparable.**

german@trigosec.com

Identify what drives the variation

For controls with multiple implementations, identify the conditions that determine which control implementation applies:

- pipeline,
- team or delivery model,
- product or service,
- technology,
- jurisdiction or risk tier.

german@trigosec.com

Example: independent change approval

> **Was each production change approved by someone independent from the change author?**

For this example, effective means:

- an approval record exists,
- the approver was not the change author.

One outcome, four approval mechanisms

The control is implemented through four approval mechanisms:

- pull-request approval,
- central release-team approval,
- change-ticket approval,
- in-house approval workflow.

> **Each mechanism can satisfy the control objective, but its evidence and potential failure modes differ.**

Does the control appear effective overall?

Outcome	Production changes
Independently approved	970
Not independently approved	30
Total	**1,000**

Across the population, the control appears to operate effectively for **97%** of changes.

Where does effectiveness diverge?

Approval mechanism	Changes	Effective approvals	Effectiveness
Pull request	350	350	100%
Central release team	300	297	99%
Change ticket	250	245	98%
In-house approval workflow	**100**	**78**	**78%**

The aggregate result concealed a materially different outcome in one implementation.

german@trigosec.com

Why does one implementation fail?

For 22 changes using the in-house approval workflow:

- an approval record exists,
- the recorded approver was also the change author,
- the in-house workflow does not prevent self-approval,
- the workflow therefore records approval without ensuring segregation of duties.

4. Evidence across systems

One control outcome may depend on evidence generated across several systems

1. Population scale
2. Controls change over time
3. Organisational variation
4. Evidence across systems

Germán Fuentes Capella

german@trigosec.com

One control question. A graph of evidence.

> **Was each production deployment authorised through the applicable process before execution?**

```
Change
   ├── authored by     →  Identity
   ├── represented by  →  Commit / artefact
   ├── analysed by     →  Quality-gate result
   ├── authorised by   →  Identity
   ├── linked to       →  Ticket / exception
   ├── resulted in     →  Deployment
   └── targeted        →  Environment
```


Translate the process into testable steps

Process step	Expected evidence
Change prepared	Change identifier, revision, author
Change reviewed	Review outcome, reviewer identity
Deployment authorised	Authorised revision, authoriser, timestamp
Change deployed	Deployed revision, environment, timestamp
Exception used	Authorisation, reason, validity period

german@trigosec.com

Map where the evidence is generated

Source	Relevant data points
Version control	Revision, author
CI/CD pipeline	Build, decision, timestamp
Ticketing system	Approval, incident, exception
Identity platform	Person, role, entitlements
Cloud or on-premises platform	Deployment, target, timestamp

german@trigosec.com

Reconstruct every process instance

- Connect records using revision, artefact, ticket, identity, and timestamps.
- Order the relevant events.
- Determine which process path applied.
- Test whether the conditions for that path were satisfied.

> **Analytics applies the same reconstruction and test across the full population.**

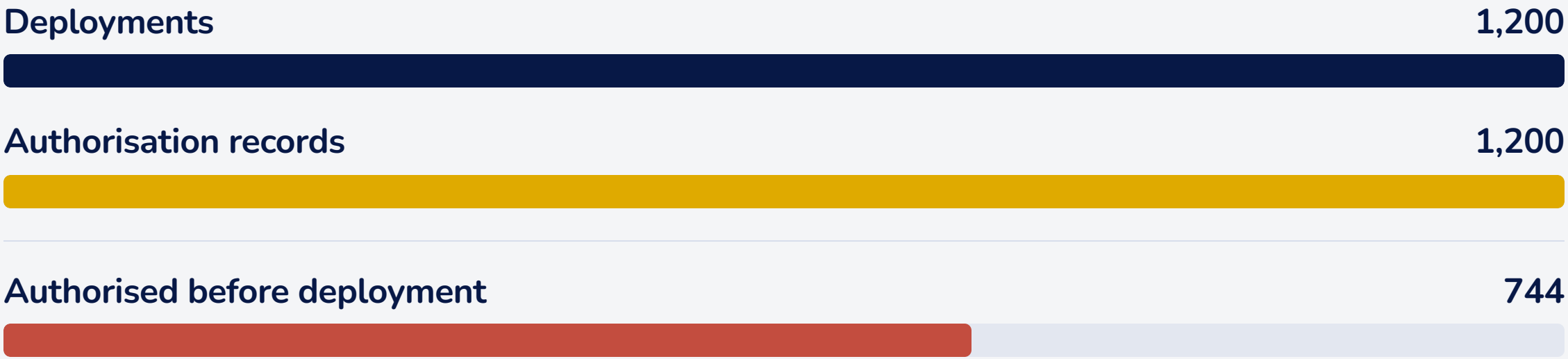
Example: authorisation before deployment

1,200 deployments. 1,200 authorisations.

Did the control operate effectively?

german@trigosec.com

Timing changes the answer



Existence testing: **100%**

Sequence analysis: **62%**

Germán Fuentes Capella

german@trigosec.com

Investigate the apparent failures

Process path	Deployments	%
Authorised before deployment	744	62%
Valid alternative: incident or emergency process	180	15%
Valid alternative: pre-authorised standard change	180	15%
Ineffective	**96**	**8%**

german@trigosec.com

What makes an alternative valid?

Incident or emergency change

Valid incident at the time; change within its scope; process invoked by an authorised role; retrospective review completed on time.

Pre-authorised standard change

Matched an approved definition or template; authorisation valid at the time; change within approved parameters; material deviations separately approved.

german@trigosec.com

Closing thoughts

What changes when we move from samples to data streams?

Germán Fuentes Capella

german@trigosec.com

Return to the four signals

Population scale

Analyse the population, then identify where to investigate.

Controls change over time

Determine whether controls operated consistently throughout the period.

Organisational variation

Compare implementations against an equivalent control outcome.

Evidence across systems

Reconstruct the process by connecting evidence from multiple systems.

german@trigosec.com

Why bring data analytics into your audits?

- Test assumptions about where risk is concentrated.
- Use population-wide patterns to direct deeper audit work.
- Reveal outliers hidden by aggregate results.
- Detect weaknesses outside the areas receiving the most attention.
- Substantiate scope and follow-up decisions with repeatable analysis.

german@trigosec.com

Where to go from here

1. Start with one control that is already proving difficult.
2. Go deep, add analytics.
3. Grow from what you learn.

german@trigosec.com

From Samples to Data Streams

Auditing organisations that change a thousand times a day

Germán Fuentes Capella

Germán sounds like "Herman"

CTO, trigosec

german@trigosec.com