Implementing AppSec programs

Tales from the trenches

WICCON 2023

POWERED BY THE BLACK CAT SOCIETY

WICCON, Oct 31st 2023

What's the plan?

- Five years of lessons learned.
  - About AppSec, DevSecOps, etc.

- Process, people and tech.

AppSec? DevSecOps?

- Any program, project or activity,
  - Aimed to teach or help developers,
  - To secure their projects,
  - In an automated fashion.

> whoami

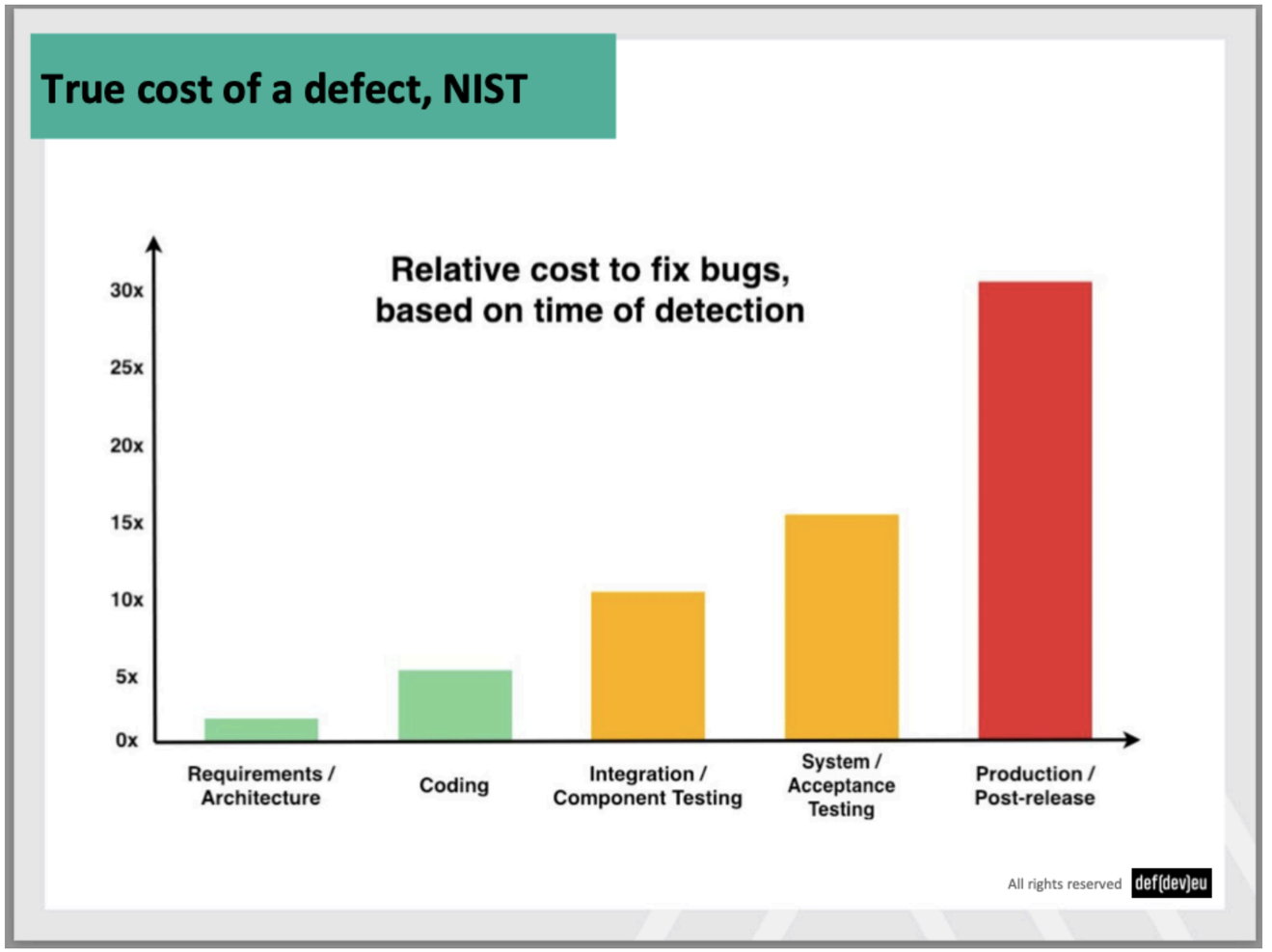
- Tess Sluijter-Stek @ Unixerius
  - *“Nerd-for-hire”*
  - *“Jill-of-all-trades”*
  - *“Eternal newbie”*

The promise

- DevOps ...
  - Will make it cheaper, faster, more efficient.

- DevSecOps ...
  - Will make it more secure, with less effort.

The sales pitch

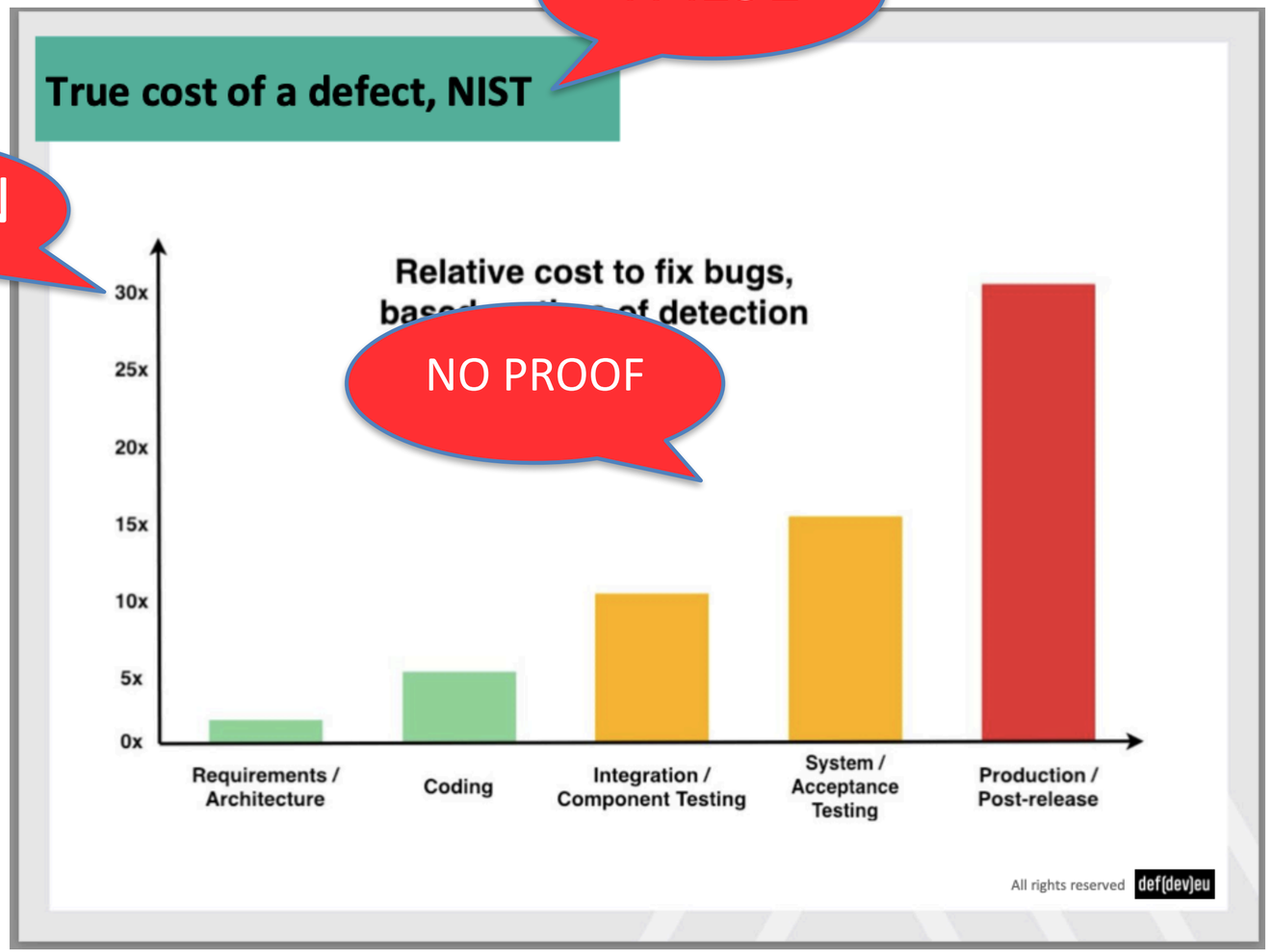


The sales pitch

FALSE

FICTION

NO PROOF



The sales pitch

- Everyone shows this graph.
  - Some say NIST, others say IBM.
  - Some cite Boehm (1981, 1976), or earlier.
  - Most state it as *Truth*®.

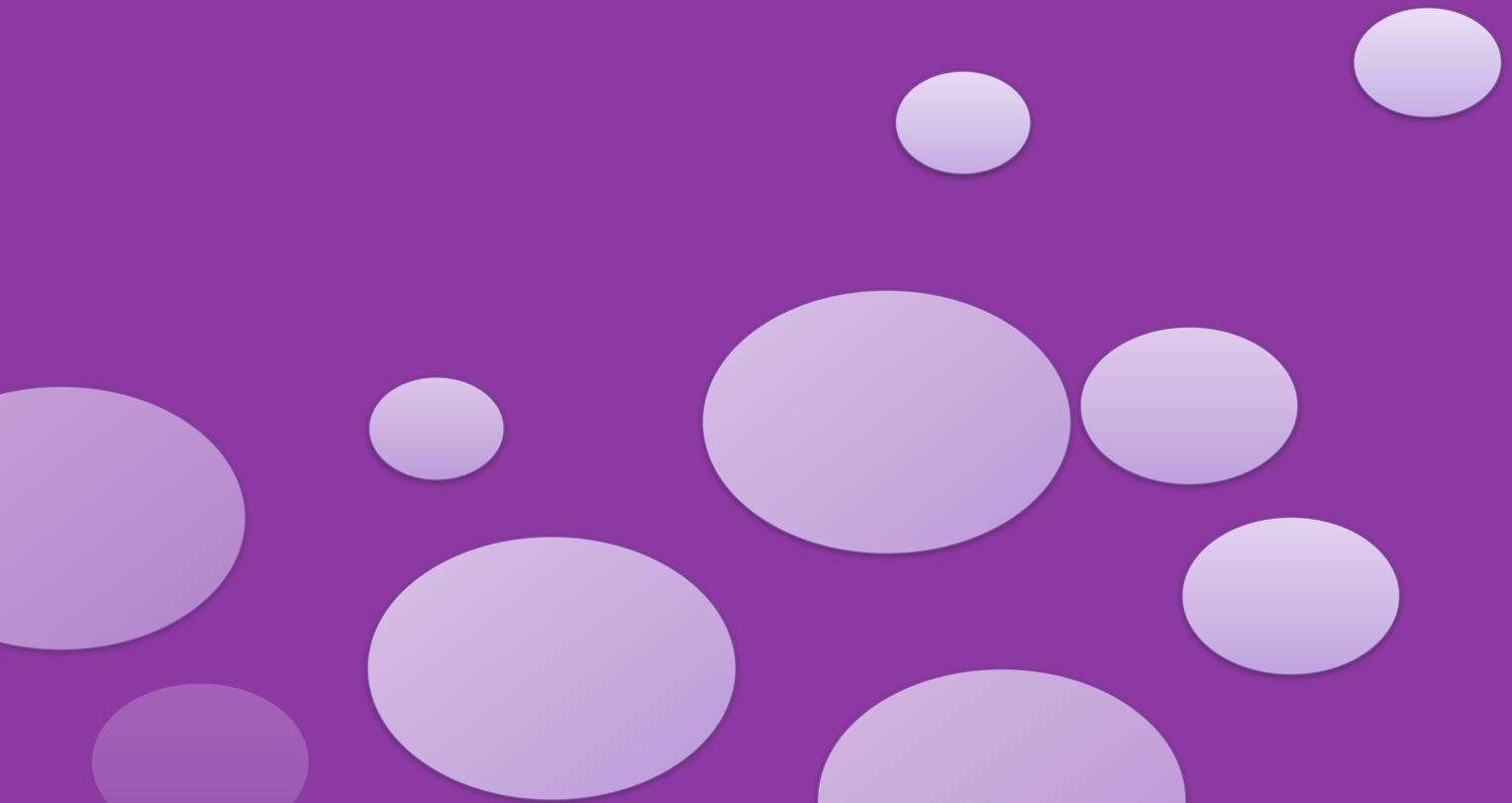
See: I ****ing hate science [H. Wayne] Leprechauns of software engineering [L. Bossavit]

Let's make this work

- "Selling" AppSec is hard,
  - Because the results are hard to quantify.

- I just want to help people deliver quality.

My experiences



Bank A

- SAST tools provided by pipeline team.
- Focus on Security Champions.
- "Certification" allows Champs to self-pentest.
- Program now under the CISO!

Bank B

- Full-blown security maturity program.
  - Maturity model & assessments.
  - In-house trainings.
  - We provided pipeline tools.
  - Lots of evangelising and social events.

Bank B - a competitor!

- A second, competing program.
  - Compulsory participation for all teams.
  - Run by external vendor.
  - Few days of training and meetings.

Government A

- Lots of confusion about goals and desires.
- Pilot program got a rocky start.
  - Resourcing, planned portfolio conflicts.

- I've snuck in trainings and socials. 😈

A quick comparison

	Bank A	Bank B	Bank B2	Govt 1
Run by	Externals	Externals	Externals	Externals
Participation	Voluntary	Voluntary	Compulsory	Compulsory
AppSec tools	No	Yes	No	Yes
In-house training	Yes	Yes	No	Yes
Vendor training	No	No	Yes	Yes
Community	Yes	Yes	Yes	Not yet
Still going?	Yes	Yes	No	Yes 😄

What did they achieve?

- <10% audience reach.
- Communities not self-sustaining.
- Nobody measured KPIs.

- But they found enthusiastic champions!

Question

- Who dislikes to be volun-told?
- Who dislikes forced activities?

- Who doesn't have time to "just" volunteer?

One lesson...

- Compulsion will not work.

One lesson...

- Compulsion will not work.
- But only volunteers won't do either.

- So evangelize,
  - And bribe.

Get a solid start

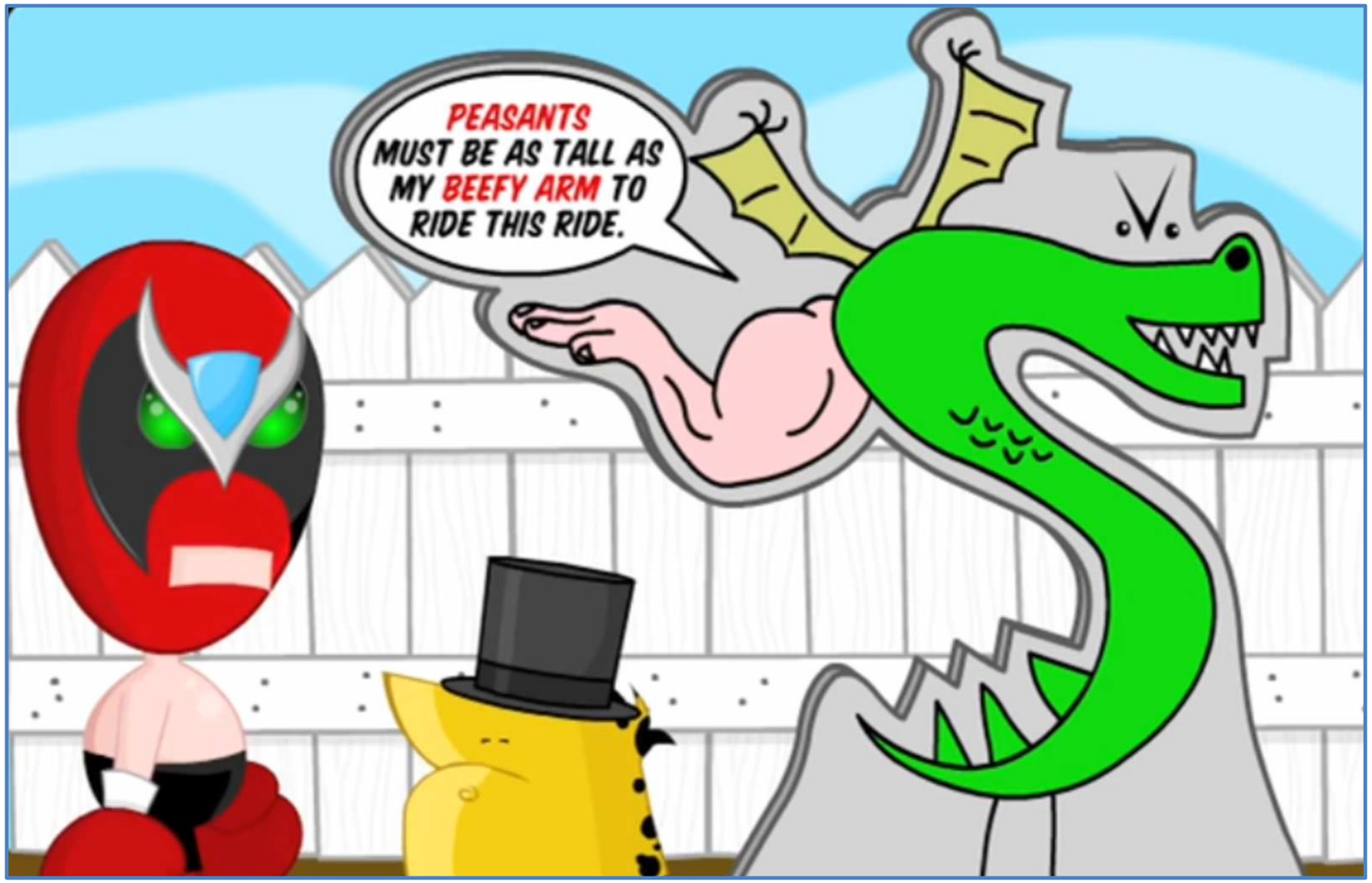
- Application Security Program Handbook
  - D. Fisher, Manning
  - Source

Process

Question

- Who knows exactly how to find their:
  - Secure coding guidelines.
  - Security requirements.
  - Explicit coding instructions.

The bad news



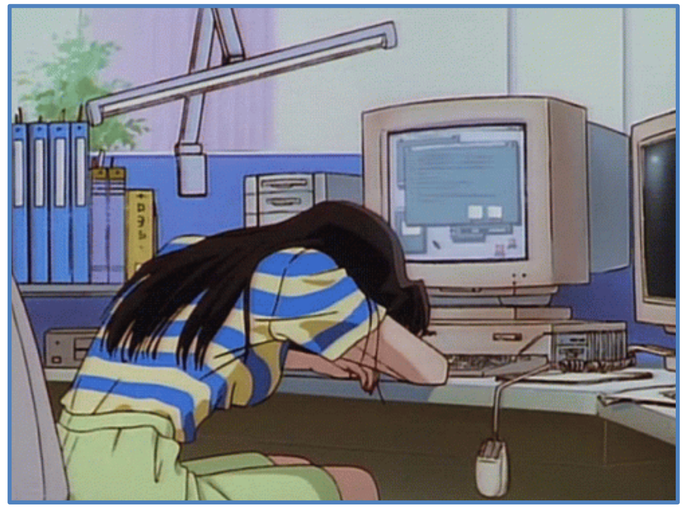
The bad news

- Your organisation needs some maturity.
  - Requirements, frameworks and standards.
  - Clarity on responsibilities.
  - Actual architects for the big picture.

Manage expectations

- Traction builds slowly!
  - The first year will feel like waste. It's not.

- Initial interest will be tiny.



First year goals

- Keep the scope limited.

- Avoid:
  - "*Solve all X, Y or Z*".
  - "*Reduce vulnerabilities by X%.*"
  - "*Reduce findings by X, Y or Z.*"

First year goals

- (Re)define processes.
- Gap analysis for tooling.
- Training materials.
- Pilot teams and projects.

Building foundations

- Define the behaviour you want.
  - Design processes to reinforce this.

- Get architects and management on board.

See: Making DevOps valuable [S. Rosenbaum]

Select KPIs, define value

- What gives value?
  - Reliability, predictability, performance.
  - Confidentiality, integrity, availability.

See: Making DevOps valuable [S. Rosenbaum]

Other metrics

- Active threat modelling.
- Completed BIA.
- MTTR for security incidents and findings.
- Automated security testing in CI/CD.

See: DevOps metrics [L.F.B. Prates]

Wait...

- That sounds a lot like maturity models?!

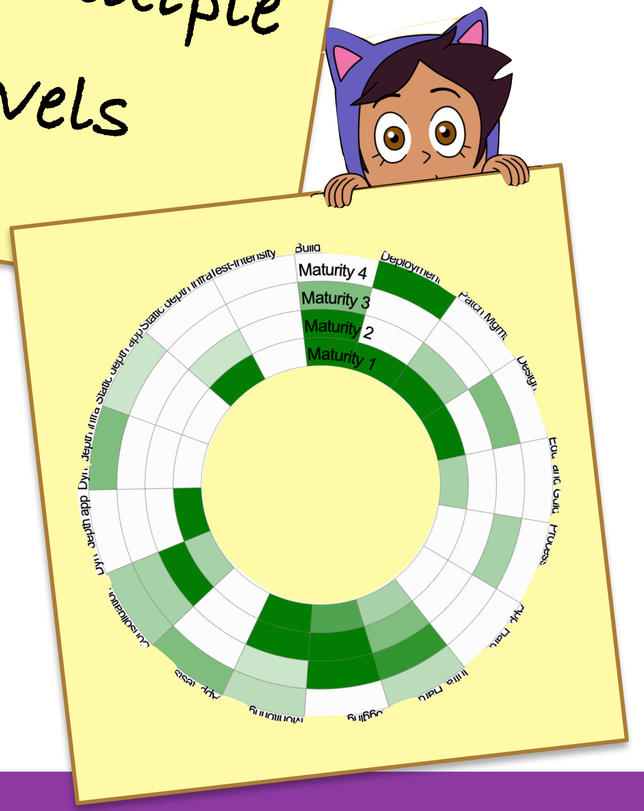
Security maturity models

- DSOM, SOMM (OWASP)
- CMMC (DoD), SANS, NIST
- ... plus commercial v
- s galore!

Each has multiple "domains".

Every domain has multiple metrics

Every metric has multiple levels



Security maturity models

- There is no wrong choice.
  - Pick, or make one.
  - Keep your audience in mind.
  - Aim for high value, low frustration.

What's in it for me?

- Cynicism will get in your way.

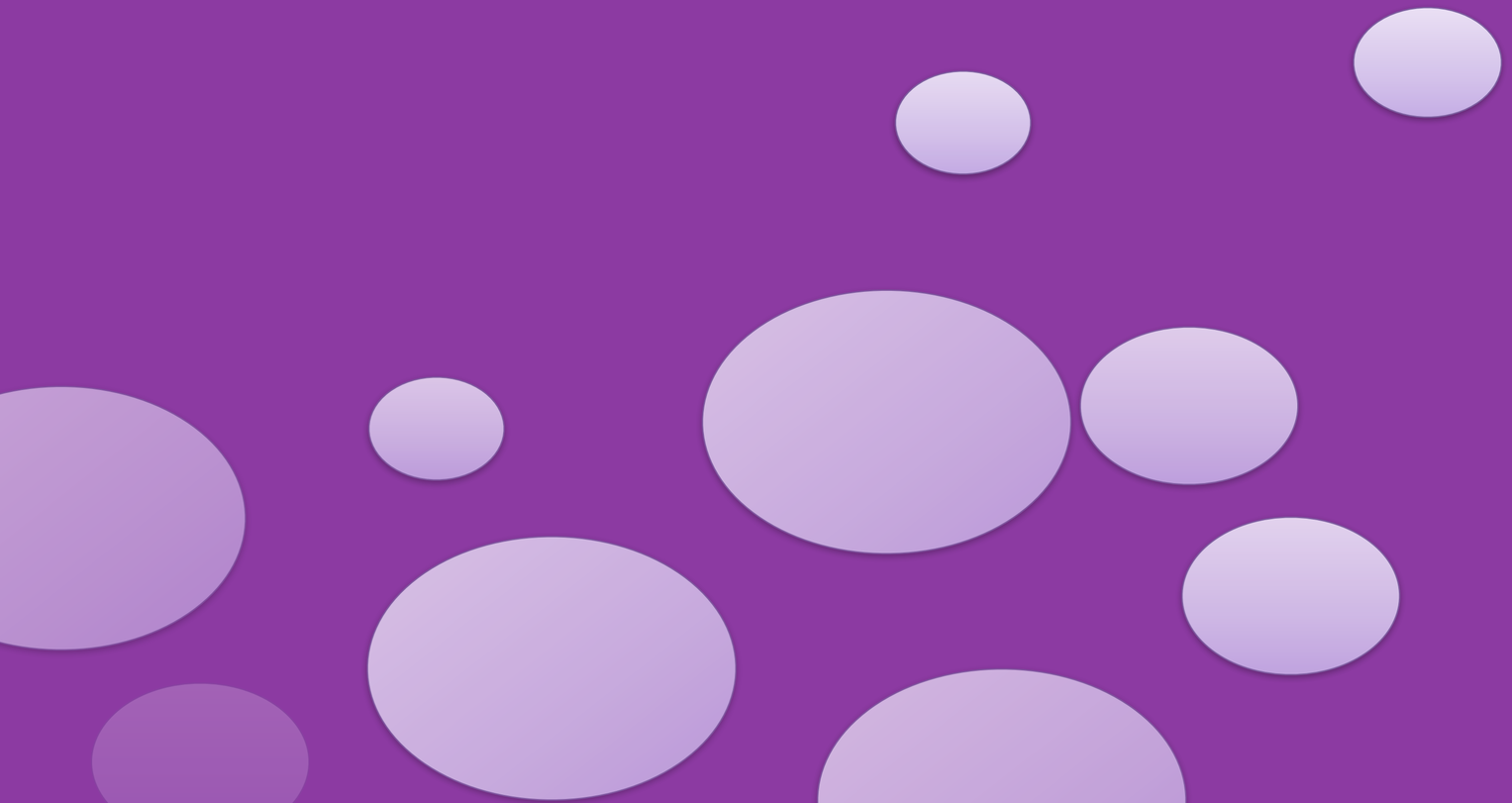
- More security findings never make anyone happy.
  - And there's no proof of speed-up, etc.

What's in it for me?

- Offer reductions in stress.
  - Priority in pen-tests.
  - Ready-to-run LCM pipelines.

- Tie-in with personal development.

People



Question

- Whose team has undergone a pentest?

Question

- Whose team has undergone a pentest?
  - That provided value …
  - That taught you something …
  - That gave actionable input …

If you're "in security"...

- Developers are real people; empathise.

- Don't offer problems and "*call their baby ugly*".
  - Offer solutions.

See: Why developers hate infosec [B. Aker]

Culture change

Image: M. Woliński

Culture change

- Your only chance is to have full C-suite backing.
  - They need to change and push policy.

- Otherwise, you go into *guerilla* mode.

Get support and mandate

- Your message will get stuck.
  - Middle management, POs, BOs.
  - Excuses.
  - Finger pointing.

Question

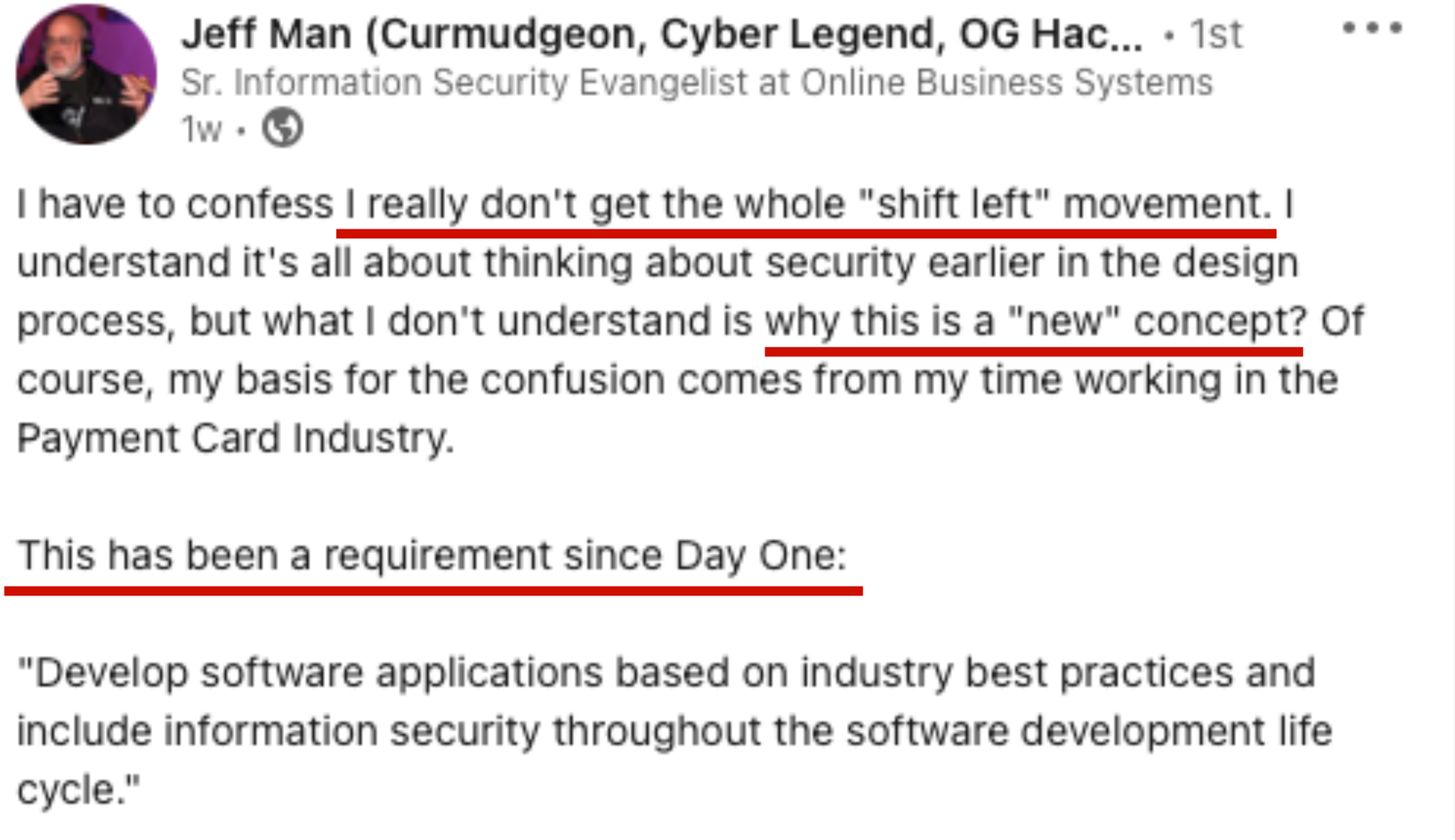
- Any developers who had formal education?

Question

- Any developers who had formal education?

- Did it teach "secure coding"?

A curmudgeon speaks



Jeff Man (Curmudgeon, Cyber Legend, OG Hac... • 1st
Sr. Information Security Evangelist at Online Business Systems
1w •

I have to confess I really don't get the whole "shift left" movement. I understand it's all about thinking about security earlier in the design process, but what I don't understand is why this is a "new" concept? Of course, my basis for the confusion comes from my time working in the Payment Card Industry.

This has been a requirement since Day One:

"Develop software applications based on industry best practices and include information security throughout the software development life cycle."

Source: LinkedIn

He's not wrong!

- I think that one challenge is:
  - This stuff still is not taught much in school.

- The other challenge is:
  - "*We have always done it this way.*"

Upskilling

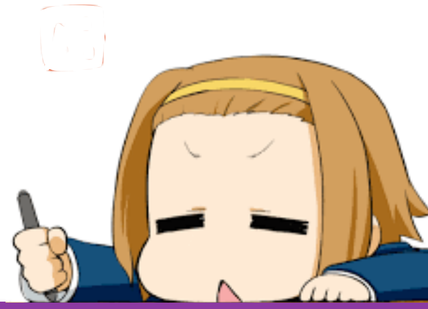
- Create your own trainings.
- Select commercial trainings.
- Fight for budget and time.
- Entice people to train.

Personnel

- Mid / Seniors:
  - "*We are understaffed and overworked!*"

- Juniors:
  - "*No-one will even give me a chance!*"

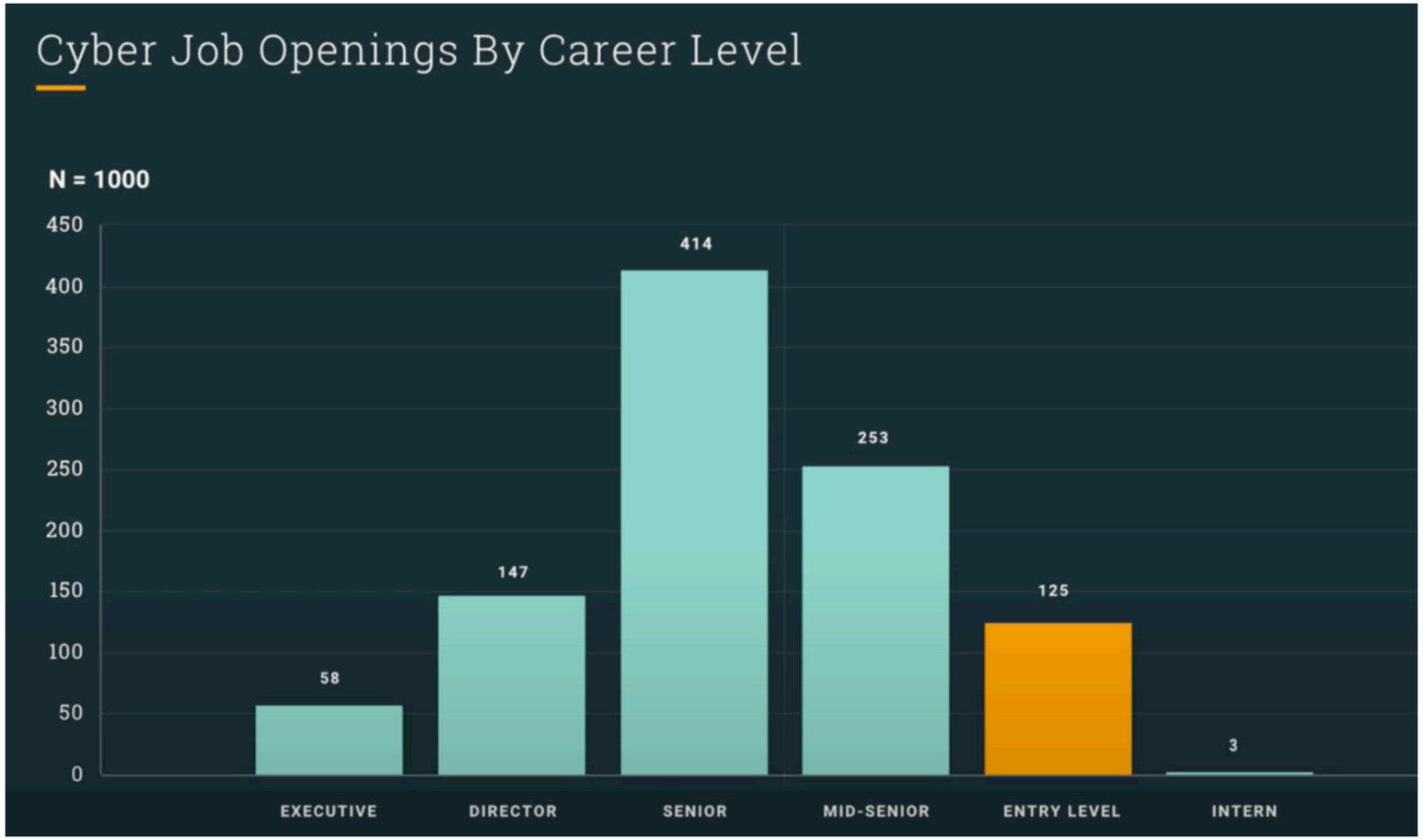
See: The dirty truth behind getting into InfoSec [N. Buckwalter]



Question

- How do *you* handle InfoSec hiring?
  - What qualifications?
  - How much experience?

Personnel



See: [The dirty truth behind getting into InfoSec] [N. Buckwalter]

Fresh hires

We claim that cybersecurity is more difficult **than it actually is.**

stop gatekeeping

See: Contempt Culture [A. Shaw]

Back to upskilling

- Train the next generation of professionals.
  - Hire fresh folks.
  - Hire career switchers.
  - Invest in them! They are (y)our future.

Celebrate success!

- Celebrate everybody's successes.
- Share on internal social media.
- Visit SIG groups.
- Tell managers.
- Have award shows!

Security champions

- Plenty companies share their success stories.
  - It's harder than it looks.

- Networking and *working-out-loud* are key.
  - Stick your nose into people's business!

Community

- In the very least, have a common chat group.
- Have community meetups.
- October CSAM.
- Security sneak-previews.
- Study groups.

Technology

CICD pipelines

- Speaking of "*you must be this tall to ride...*"
  - If this is missing, you're in trouble.

- Automation is key to efficiency.

Image: Potsatou

CICD pipelines

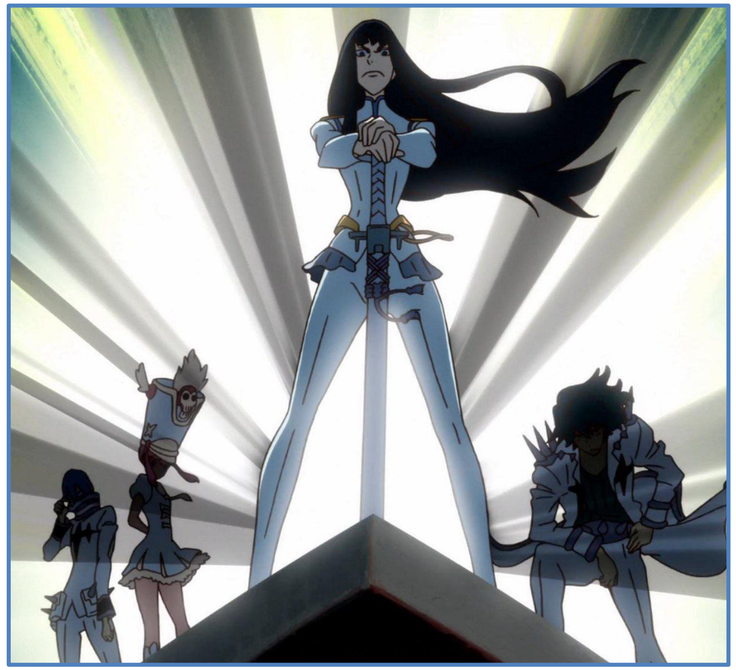
- Provide ready-to-go solutions.
  - Pipeline templates.
  - Do all the SAST, SCA, secrets, etc.
  - Automatic LCM.

- With reporting in one central spot.

Picking your tools

- Requirements and standards apply to us too!
  - Don't introduce *shadow IT*.

- Don't work alone, but ...
  - Beware *death by committee*.



When you find legacy deps

Life is hard, legacy is harder

- We may think, "*Come on, just update your ...*"
  - *pom.xml, package.json, requirements.txt ...*

- Tools like *Renovate* can help, but aren't magic.

See: Why developers hate infosec [B. Aker]

Closing

Take aways

- Agree on clear goals, manage expectations.
- Get full C-suite commitment.
- Make friends, "pull up", empathise.

Thank you!

- It's been awesome!

Reference materials

Resources

- I ****ing hate science!
- Degrees of dishonesty
- The leprechauns of software engineering
- A short history of the cost per defect metric
- Application Security Program Handbook
- Making DevOps valuable

Resources

- DevSecOps Metrics
- LFD-121 Developing secure software (free)
- Contempt Culture
- The dirty truth behind getting into InfoSec
- Why developers hate infosec