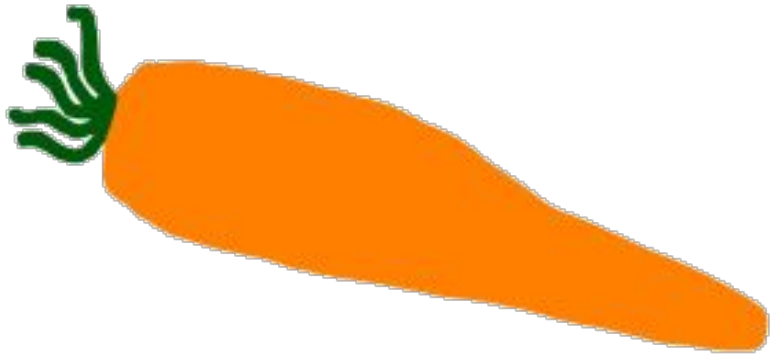


Being Technically Correct isn't Always Helpful

David Robinson
Karit

OWASP AKL
September 2025

#whoami

- David Robinson
- Karit
- Do SecurityStuff™
- Wellington
- Kākācon
- https://karit.nz/social

Today

- Some examples of poor advice
- Some ways to make it better
- What to do if you get bad advice

Being Correct

Correct Security Answer

- Often there is correct security answer
- This is often not particular for all situations
- Need to understand the situation

Is this actually this bad?

- Why do we say not write down?

https://www.zazzle.co.nz/minimalist_yellow_gold_stripes_password_notebook-256131491108192619

Don’t write passwords

- Seems simple
- Got to look at the bigger picture
- In office yeah don’t write down
- But at home? (assuming no DV etc)

What are they doing if they don’t write down?

- One password everywhere
- Super vulnerable to password stuffing
- Would it be better if they write passwords down but use unique passwords?

Being “correct” and perfect

- Can be overwhelming to some people
- So they give up and do nothing

Pen Testers

Pen Test Reports

- Reports lacking:
  - Context and reasoning for a vulnerability type
  - What the issue means to the business

Non exhaustive list

- These are just a small set of examples
- These are just here to get you thinking about the concepts presenting here

Server Version Number

- CWE-497: Exposure of Sensitive System Information to an Unauthorized Control Sphere

Azure Load Balancer

- Has a cookie called:
  - ApplicationGatewayAffinityCORS

SaaS Products

- All have their own cookies
- HTML element names

Have seen

- This called out for information disclosure.
- Technically it discloses that use an Azure Load Balancer
- Or name of SaaS product

But…

- Think about the what is the root this Vulnerability Class is targeting?
- Know what software running to use known issues against
- But with SaaS customer not running

Customer

- Can’t patch
- Trust SaaS provider to patch

Also

- Given the IP ranges used know the provider anyway

Including this

- Shows that you don’t understand the vulnerability class

Also remediation steps

- Often see generic steps
- Not customised to the product
- Make sure include steps/references that are for the platform in question
  - AWS, Azure documentation not a generic web page

Why?

- Show that you actually know what the platform is and not just reporting generic vuln scans
- With generic text

Session Logout

- CWE-613: Insufficient Session Expiration
- Don’t get logged out serverside when click logout

Standards have changed this

- Standards and way of doing things change
- Single Signon and JWT
- Logout doesn’t work the same way anymore
- Single Signout

When you call this out

- Make sure you understand the pattern and limitations
- WIth JWTs there isn’t a server session anymore
  - And there are multiple servers involved that don’t share state
- The design pattern is now time expiry not logout

Single Signout

- There are now design patterns for Single Signout
- Though not all SSO providers have the required APIs
- Even less apps implement the hooks for single signout

What to call out

- Don’t call out the logout button not logging out immediately
- Check that refresh tokens don’t reissue
- Understand the design and pattern in use
- State if the IDP and App actually supports Single Signout or not

TLS

- Yes they server might accept some weak ciphers
- Most customers have no control over TLS ciphers in 2025
- Most organisation are using a CDN or a load balancer
- All as a Service

Customer can’t change

- Unless you can point at vendor documents that says what settings the the customer can change
- Why are you reporting it?
- Realistically they aren’t going to get pwned
- You are just creating noise for the customer to deal with

If you really must include

- Do the leg work
- E.g. say why it is included and that no action possible
  - You have a NZISM requirement
  - TLS cipher X means Y
  - Up to vendor to change, no action for you

XSS

- Reflective XSS
- Self XSS has a history of being stunt hacking and not helpful to raise
- But with context it can be important

PCI as context

- PCI has some particular requirements around JS interfering with the Card Payment flow
- If you can show it having impact on PCI
- It becomes much more applicable to the business
- Showing this can go from an ignore to a fix
- Understanding the business context is important

Standards

- Have seen reports that say need to do X else not won’t be in compliance with a standard
- Did the customer say they require that standard?
- If not, they don’t care about that standard
- Sure if there is a business impact can use the standard as a justification for recommendation

Frame for the business

- Frame the finding for the impact on the business you are engaged with
- Say why the finding is important to the business

Remember the report

- Is what the customer gets for a lot of money
- Make it helpful
- Talk about the issues that matter
- Don’t make noise
- E.g. Daniel Ayes and JS libraries in 2021
    - Causes damage and lost of trust

Reproduction Steps

- Should be how to reproduce the issue described in the impact
- If your reproduction only show the potential of an impact, make sure your description says that

Don’t cut corners

- Don’t cram too much into one finding
- Having multiple issues
- Confuses the description and impact
- Harder for the customer to understand

Multiple Remediation

- If there are multiple options in remediation
- Often only gets done or only goes to one team
- When different things to fix make it clear
  - That an AND
  - Better yet if different issues they are different findings

Blueteam

Problem Statement

- What can you do if you are getting this less than helpful advice?

What are you communicating?

- Tell what standards, compliance frameworks, etc you care about during scoping
- “What is keeping you up at night?”
- Give them context for the engagement

Make sure output format is useful to you

- Are they just giving you PDF
- Hard to copy and paste from
- Can’t add tracking detail
- Get a spreadsheet along side the PDF
- Most places are using a report generator, so should be no effort to provide a spreadsheet

Ask questions during the close out

- You can ask questions
- If references are generic, ask them for references that are for your platform
- If they give you findings that aren’t applicable, tell them that

Wrapping it up

Pentester

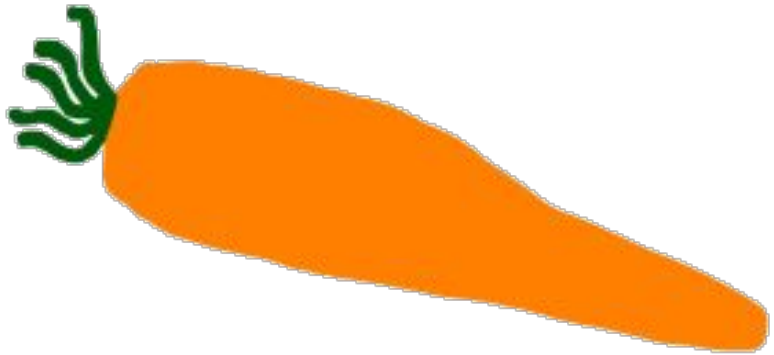
- Are the findings actually useful?
- Are the findings actionable?
- What is core point of the vulnerability class?
- Make sure it isn’t just noise
- Does the remediation suit the business needs?
- It is only helpful if it is actionable
- Spreadsheet not just PDF

Be Specific

- References for the platforms, libraries, etc in use
- Remediation steps for the platform not generic
  - Not: Turn off TLSv1.0
  - Say how to do it

Blueteam

- Are you asking for the right things?
- Tell the them when it doesn’t make sense



Thanks

https://karit.nz/social