
Cigital
BUILDING SECURITY IN
The 7 Myths of Software Security Best Practices

The 7 software security myths are common misconceptions about software security best practices.

They explore how software security initiatives (SSIs) should work, and aren't simply about how to secure a particular application.

Cigital

Myth #1

Perimeter security can secure your applications.

Cigital

The Truth

Perimeter security was originally implemented to deter and block disturbances from the outside. Although this is still a good basic security precaution, attack strategies have become far more advanced.

Cigital

Let’s Put It Into Perspective

Today’s computer and network security mechanisms are like the city walls, moats, and drawbridges of feudal times.

Cigital

Feudal Systems Are a Thing of the Past… For a Reason.

- Attack strategies have become more advanced in the past 500+ years.
- A moat, city wall, and drawbridge are no competition for those who want to break in.
- Today's attackers have access to things like predator drones and laser-guided missiles!
- The modern castle requires more protection against attacks.

Cigital

The Solution: Move Beyond the Perimeter

Due to more advanced threats, we must make sure our modern day castles are protected by more than just perimeter security measures.

Learn more about myth #1

Cigital

In the beginning, perimeter security was invented to protect this broken stuff from bad people.

But, more importantly, why is the stuff broken in the first place?

Cigital

Myth #2

A tool is all you need for software security.

Cigital

The Truth

Tools aren’t a catch-all solution. Although more advanced tools allow new rules to be added over time, if a rule hasn’t yet been written, the tool will never find that problem.

Cigital

Let's Put It Into Perspective.

- Black box testing tools were the earliest approach for simple protocols (such as HTTP).
- Next came the idea of looking at the code itself with static analysis tools.
- Static analysis tools focus on finding implementation bugs.

However…

Cigital

The best a code review can uncover is about 50% of security vulnerabilities.

Cigital

3 Reasons Tools Aren't a Catch-All Solution

1. Tools suffer from false negatives and false positives. False negatives are more dangerous because they lead to a false sense of security.

2. Manual auditing is very time consuming. There's no way to avoid sorting through the output, prioritizing which issues.

3. Tools still need people. There's no way for a tool to know which problems are more or less important to the organization.

Cigital

The Solution: The Happy Hybrid Approach

- Use the tools. They expedite processes and eliminate the low-hanging fruit vulnerabilities.

- Don’t underestimate the effectiveness of humans.

Learn more about myth #2

Cigital

Myth #3

Penetration testing solves everything.

Cigital

The Truth

Penetration testing is the most frequently and commonly applied of all software security practices.

But, this isn't necessarily a good thing…

Cigital

Let’s Put It Into Perspective.

- If we characterize functional testing as “testing for positives,” then penetration is “testing for negatives.”
- Testing for a negative poses a far greater challenge than verifying a positive.
- If negative tests don’t uncover any faults, this is only proof that no faults occur under particular test conditions, and by no means proves that no faults exist.

Cigital

The Solution: Outside-In Penetration Testing

- Conduct penetration testing at the end of the SDLC to ensure security measures introduced earlier in the lifecycle are effective.

- Architecture risk analysis and code review should take place before penetration testing.

Learn more about myth #3

Cigital

Myth #4

Software security is a cryptography problem.

Cigital

The Truth

You can use a crypto library to add a security feature to an application, but that's not the same thing as making an application secure.

Cigital

Let’s Put It Into Perspective.

Security is a system property, not a thing. Simply adding crypto measures to your code is unlikely to make it secure.

Crypto:

- Is astonishingly complex to get right.
- Can’t find or eradicate bugs and flaws.
- Can’t train your developers.
- Falls prey to penetration testing from time to time.

Cigital

2 Reasons Crypto Alone Is Not the Answer

1. Most attacks against applications do not target security features specifically (and crypto is just another security feature).

2. Applied crypto is designed to protect data, not the application that processes data.

Cigital

The Solution: Focus on Practices, Not Features

Software security is about integrating security practices into the way you build software, not simply integrating security features into your code.

Learn more about myth #4

Cigital

Myth #5

Software security is only about finding bugs in your code.

Cigital

The Truth

Implementation bugs in code account for 50% of the overall software security problem.

Cigital

Let's Put It Into Perspective.

- The division of design flaws and bugs is about 50/50. Both need to be secured.

- You can institute the best code review program on the planet, with the strongest tools known to humanity, but it's unlikely that you will be able to find and fix flaws this way.

- Flaws must be found and resolved by experienced people.

Cigital

The Solution: Find, Fix, and Block Bugs

- While finding bugs in your code is great, however to improve your security, it's critical to fix what you find.

- It's also important to make sure you train your developers not to make new bugs every day.

Learn more about myth #5

Cigital

Myth #6

Software security should be solved by developers.

Cigital

The Truth

The notion that developers—and only developers—should collectively and magically be responsible for software security never actually works in practice.

Cigital

Let’s Put It Into Perspective.

Who should ‘do’ software security?

- ❑ Security people
- ❑ Compliance people
- ❑ Developers
- ❑ None of the above
- ❑ All of the above

Cigital

The Solution: Creating an SSG

- Software security does (eventually) take **all of the above**, but ultimately must be coordinated by a central software security group (SSG).

- Implementing software security requires forming a SSG.

Cigital

We asked 23 real firms how they went about organizing their SSG.

5 Essential SSG Groups

1. Services
   - Structured in terms of specific services provided to organization.
2. Policy
   - Sets policy that is to be enforced by technologists outside the SSG.
3. Business Unit
   - Tailor solution to each business unit within the organization.
4. Hybrid
   - Sets mandatory policy, but doesn't have bandwidth to execute work for all teams who want it.
5. Management
   - Very mature initiatives managing a distributed security network.

Cigital

Tips to Make Your SSG Successful

- Make sure your SSG includes software people, security people, and people who can interact with the business.
- Train your developers and arm them with software security know-how.
- Check code for bugs, check design for flaws, and check systems for security holes.

Cigital

Myth #7

Software security should be solved by developers.

Cigital

The Truth

Getting started back in the day meant identifying apps with the most risk and focusing all of the attention on them.

However, those days are over.

Cigital

Today it's about securing the entire portfolio—the overall attack surface.

Cigital

Let’s Put It Into Perspective.

- Risk management is a careful balance between security and business functionality.

- Unfortunately, risk management can fail, and when it does, it *really* fails!

Cigital

Failure 0: An Exposed Portfolio

- Security controls are kind of like clothing for software and you have a large portfolio of “naked” software assets to protect.
- Determine how many, and which type, of clothes to pair with your software assets.
- The types of risk management decisions made for this prioritization makes all the difference between success and failure.

Cigital

Failure 1: How Many Medium Risks Make High Risk Software?

- There is a bug severity problem in software.
- How many little bugs does it take to make a big bug?
- The problem of compounding influences security risk and risk management decisions.

Cigital

The Solution: Cover Your Entire Software Portfolio

Ultimately, both risk management failure conditions can be addressed by the same guiding principles:

- Leave no code naked.
- Tighten security controls between low and high risk assets to the extent possible without moving the high risks down the prioritization list.
- Create a specific testing schedule.

Cigital

The Bottom Line

- Clearly, software security is about more than point solutions.
- No point solution by itself can solve software security.
- A successful SSI is about culture change in an organization from top to bottom and back again.

Cigital

For more info about real-world SSI activities and how to go about measuring an SSI,

go to https//www.BSIMM.com


info@cigital.com


@cigital

Cigital