Survivability Engineering, Part 1: Risk Assessment

The Security Brutalist. securitybrutalist.com

Intro

You are trying to understand three things, in this order:

- What can actually happen here?
- If it happens, what really breaks?
- Are we currently positioned to survive it?

Step 1: Why does this exist?

- What business function does this support?
- Who depends on it?
- What stops if it stops?
- How long can the business tolerate it being down?

Step 2: How could it actually be compromised? (Assume a realistic attacker, not a movie villain.)

- How is it exposed?
- Is it reachable from the internet, user devices, or other systems?
- Who can log into it?
- What other systems trust or integrate with it?
- What is the shortest believable path to control?

Step 3: Assume compromise. What can the attacker do?

- Can they access sensitive data?
- Can they modify transactions or configurations?
- Can they create new privileged access?
- Can they pivot to other critical systems?

- Can they take it offline?

Step 4: What really breaks?

- What operational processes stop?
- When does revenue feel it?
- When do customers notice?
- Does this trigger legal or regulatory exposure?
- What is the realistic downtime?

Step 5: Can we survive it?

- How fast would we detect it?
- How fast could we contain it?
- How fast could we restore it cleanly?
- Have we proven this, or are we assuming?

Step 6: What single change most improves the outcome?

- What reduces blast radius?
- What shortens recovery time?
- What limits pivoting or data loss?
- What is achievable quickly?

(Optional) Step 7: Add $ if needed

- Downtime (days) × business cost per day = rough $ exposure
- Compare to cost of improvement