

Product Security Incident Response at a Fortune 500 SaaS

Garrett McNamara

You are in the right room



Garrett McNamara

Former: dev, researcher, educator



NOW

- Sr. Product Security Response Manager, Founder of ServiceNow PSIRT
 - CNA
 - FIRST.org
- MBA student

BEFORE

- CNA x2; FIRST.org x1
- Okta
- Forcepoint PSIRT
- Gov contractors
- Invincea / Sophos
- Search and rescue volunteer (for fun)

Mostly hosted



Mostly on-prem

Garrett McNamara

PSIRTs since 2015*

- *18-month break as an AppSec educator working with devs @ Okta



Type 2 fun enthusiast

- You all are too fun
- Can't escape PSIRT
- PSIRT is life



Premise

- 💡 Product security incident response at a SaaS technology company comes with **challenges and opportunities different from those at a strictly on-prem vendor.**
- ✗ Challenges include easily discoverable and often wide-open Internet connected attack surface area.
- ✓ Opportunities include that rapid risk-based decision-making is enabled by the ability to measure exposure at scale and monitor for exploitation activity.

Agenda

YES ✓

- Risk factors
- Hosted vs on-prem
- Challenges (~70%)
- Opportunities (~30%)

NO ✗

- Advice
 - + I'm not a lawyer
 - + Views are my own
 - + Your needs may vary
 - + My advice is very bad

Risk Factors

- Speed of attack surface discovery on shared infrastructure
- Colocation / subdomains can mean easy enumeration
 - ...and accidental overspray
 - Not suggesting you rely on obscurity!



Dave Dugal?

Risk Factors



- Fast researcher ramp up:
 - [Opinion] Web tech has a lower learning curve for researchers to find at least basic vulnerabilities
 - Accessible (i.e., free), instantly ready
 - Minimal hardware investment
- Easy target access:
 - Internet connected / no customer-controlled network isolation / less defense in depth
 - CVSS scores tend to start higher due to Attack Vector (AV) == Network
 - Ingress and egress requirements / can't interfere / shared infrastructure

In other words...



Welcome to the show

Cloud doesn't always mean hands off

Using a hosted / cloud vendor doesn't necessarily remove all customer involvement:

- Shared responsibility model
- Customer risk decision making
 - Patch now or later
 - Apply mitigations
- Unclear expectations in time of crisis



Challenges - Visibility



Potential customer surprises after a vulnerability disclosure:

- Vendor may lack visibility **by design** into requests and responses (weighing privacy concerns).
- Vendor therefore cannot advise on whether a data leak occurred.

Challenges - Mitigations

- Mitigations (WAF) can break functionality for all or even just some customers
 - Some customers would rather endure some downtime than data leak
 - How much downtime until permanent remediation
- Rate limiting can vary by use case
 - Power users use cases may break (bulk downloads / rapid API calls)
- Hosted providers do not have unlimited capacity against DoS

- Malicious traffic doesn't always look different



Challenges – Disclosures

- CVEs for cloud if no action required?
 - If auto-patching enabled, was action required?
 - Customer enablement could still come in the form of manual patch adoption faster than scheduled
- How soon to publish?
 - Give customers time to patch before full CVE details released; but
 - Some do not act unless vulnerability management tooling flags for a CVE
 - Bonus: do any customers expect warning before others?
- At thousands of customers (each having 1 or more staff), embargo is complex

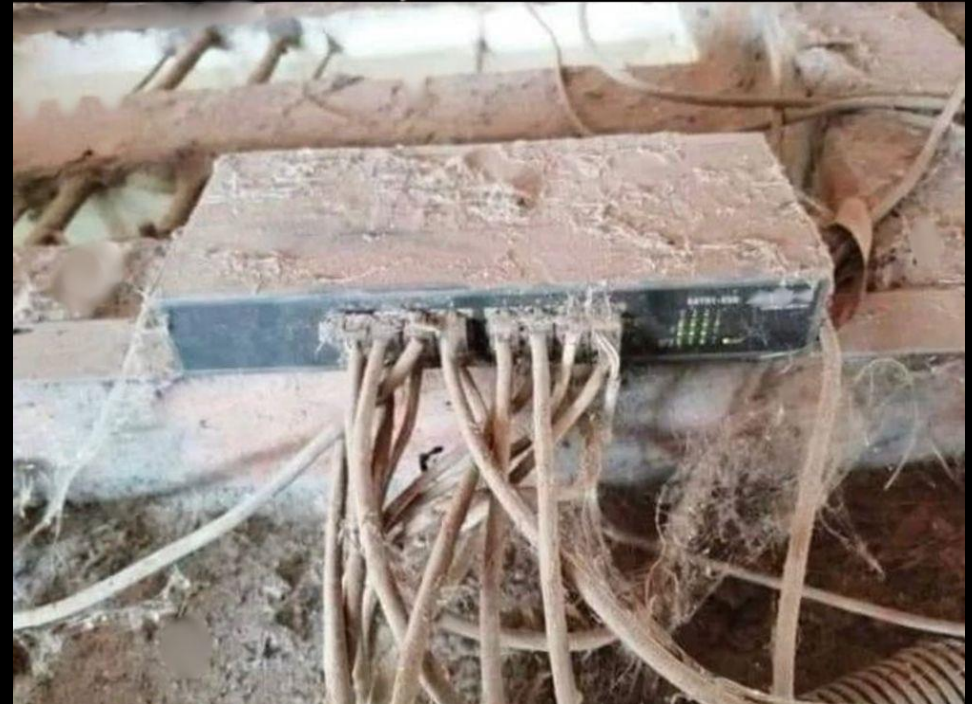


Challenges – Intentions

Did the customer intend to do that?

- Do they *mean* to have that set up?
- Do they *know* they have that set up?
 - Did someone ten years ago who later quit set it up?
- Have people built on top of the convenient problem without knowing it?
 - *It just works*

If it works, don't touch it.



Challenges – Intentions, part 2

- Breaking changes
 - Three ring model:
 - (Vendor) Platform behavior (PaaS)
 - (Vendor) Re-use of that behavior to make apps (SaaS)
 - (Customer or partner) Also using that behavior (custom code)
 - Which means, multiple dev audiences to educate
- Signature mismatch on modified files / too dangerous to touch?
- Arbitrate abuse of other Internet services
 - Don't want your shared infra to be banned

Challenges - Enablement

Shipped secure, but option to reduce that still lands vendor in the news.

No win situation.

Yes, responsibility on customer but it's a dead right situation in the court of public opinion.



Challenges - Comms

Comms failures

- Expired customer security contact info
- PTOs without coverage
- Security and maintenance and consumer teams may be different
- Relay failures with managed providers



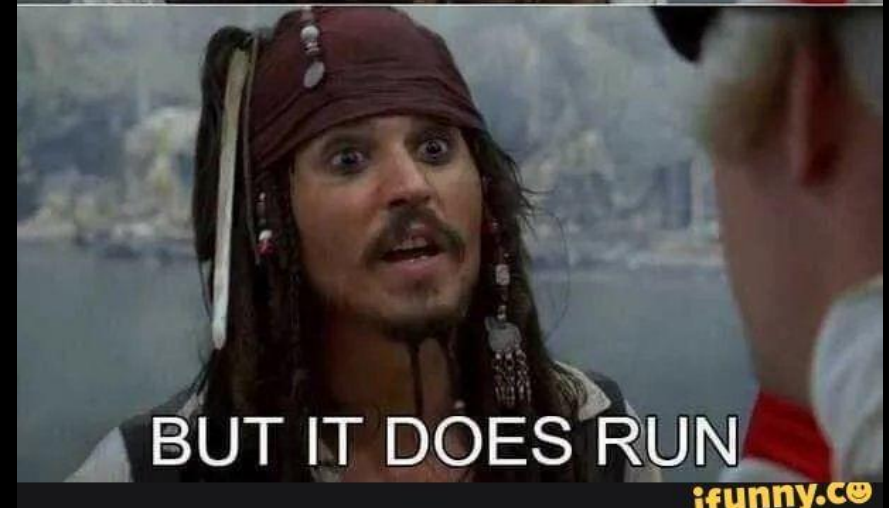
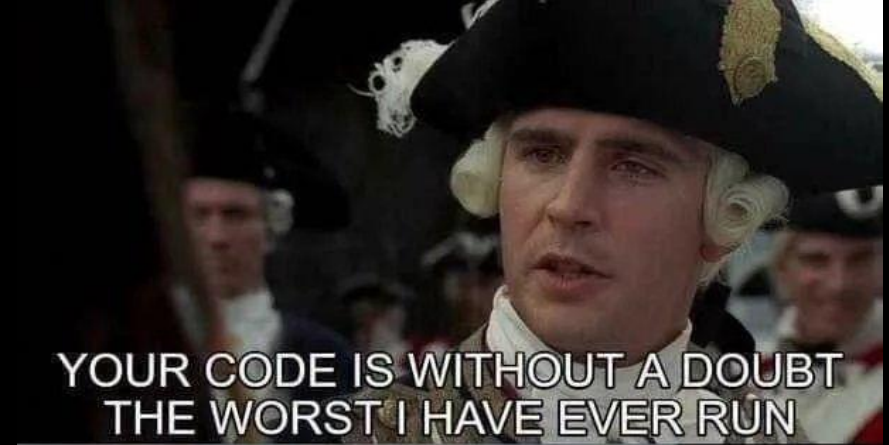
dear slim

i wrote you but you
still aint callin

Challenges - Features

Living Off the Land (LOTL)

- At vendor expense especially if the software has powerful features. These can include the abuse of other services.
- Even if not malicious, just poorly written custom code.



Challenges – Maintenance

- More stuff! Hosted providers are responsible for addressing vulnerabilities in the entire software stack
- But wait... even a small percentage of on-prem business means the product (and its security patches) are still subject to reverse engineering
 - Tactically acquired



Challenges – Steering

- Block ability for rollbacks in underlying platform software.
- Revoke vulnerable versions from app store.
- Soak time for testing changes- how much to allow? May have customers who only want to update yearly. Researcher wants shorter timeline- e.g., 90 days.



Opportunities – Hosting's not all bad?

- Honeypot gathering opportunity.
Even if it's infrastructure that wasn't meant to be a honeypot.
- Get a data set for sale and realize it's junk. Judgement call:
 - + Do you buy it? Do you report out that it's junk demo data? Does it matter?
 - + Ensure even demo environments are patched with same urgency as real environments



Opportunities – Observing

- Being sane about what to escalate to accelerate remediation SLA
- Observed testing activity in common across customers = suspicious = blocking
 - + Watching for proof-of-concept maturity evolution
- Ability to measure true exposure quickly:
 - + Versions adopted
 - Component adoption
 - + Relevant configurations
 - + Prod vs subprod deployments
 - + Quantity of data in use for xyz component
 - Some components come with demo data

Opportunities – Accelerating

Ability to force change or urgent comms, if needed

- Secured right away, but with downsides:
 - Disruptive to everyone
 - Establishing precedent overextending in the shared responsibility model
 - Difference in customer preference on breakage vs locking down.
 - Breaking may just change impact from Integrity and/or Confidentiality, to Availability.
 - Does breaking something count against uptime guarantee?

We talked about

PSIRT at a SaaS has challenges and opportunities different from those at a strictly on-prem vendor.

Thank you,

Garrett McNamara

garrett.mcnamara@servicenow.com

