Information Security IRL

IMPLEMENTING SECURITY AS AN ENGINEER IN 2020

MAXIMILIAN BURKHARDT • APRIL 13, 2020

A Very Brief Intro

UC Berkeley ➡️ iSEC Partners (pentesting) ➡️ Airbnb (defense)

Agenda

1. Why we need people like you!
2. What is security?
3. Making security happen
4. Real talk: a security incident at Airbnb
5. Places to go with your infosec career

Some XSS History

20 YEARS AGO

2 CA-2000-02: Malicious HTML Tags Embedded in Client Web Requests

This advisory is being published jointly by the CERT Coordination Center, DoD-CERT, the DoD Joint Task Force for Computer Network Defense (JTF-CND), the Federal Computer Incident Response Capability (FedCIRC), and the National Infrastructure Protection Center (NIPC).

Original release date: February 2, 2000
Last revised: February 3, 2000

A complete revision history is at the end of this file.

Source: https://resources.sei.cmu.edu/asset_files/WhitePaper/2000_019_001_496188.pdf

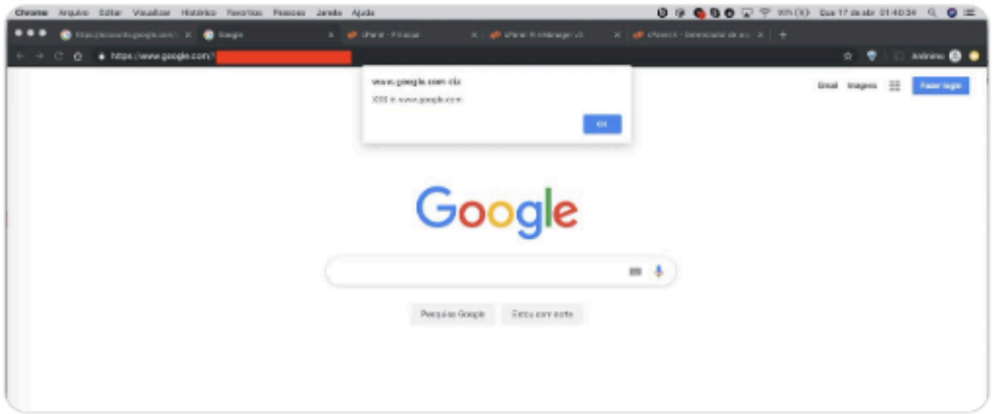
Some XSS History

JUST LAST YEAR

My first tweet.

A sweet XSS in Google main page and almost every subdomain.

=)



10:59 PM - 16 Apr 2019

Source: https://twitter.com/WHHackersBR/status/1118393568656334850 (not necessarily reputable)

What's the Deal?

- Google has one of the best security teams out there
- From 2015-2016 they paid out $1.2 million for XSS bugs (via bug bounties)

Source: https://security.googleblog.com/2016/09/reshaping-web-defenses-with-strict.html

Security isn’t scaling

But everything else is.

It’s Not Just Old Problems

WE KEEP THINGS INTERESTING

- Containerization / Kubernetes is bringing new problems (and opportunities)
- Blockchain?
- Crazy dark magic: Spectre, Meltdown, Rowhammer

So Why Get Involved?

IT'S NOT ALL FIRES

- There are huge opportunities for changing how the industry does security
- We get to work at the bleeding edge

Security is Creative

- Different tech stacks
- Different threat models
- Different budgets
- Different company cultures

What is Security?

“A system is secure if it behaves precisely in the manner intended — and does nothing more”

— Ivan Arce

... which is not very helpful.

What is Security?

AN ATTEMPT AT MORE USABLE DEFINITIONS

- It’s a strategy to address risks to your system
- It’s all about defining what the threats are and responding appropriately

How to Mitigate Risk

IN THE BROADEST POSSIBLE TERMS

- Be threat-agnostic
  - Build protection close to the assets
  - Assume some defenses will fail
- Be ready to detect when defenses fail, and be able to respond
- Self-assess constantly
  - Human review is still really useful!
  - Bug bounties are great at this too

So How Is This So Difficult?

HONESTLY

- What's the hangup?
- "In theory, there's no difference between theory and practice. In practice, there is."
- Modern information systems are built on growth, and if security opposes growth, it won't happen.

Making Security Happen

We're Pretty Good at Incident Response

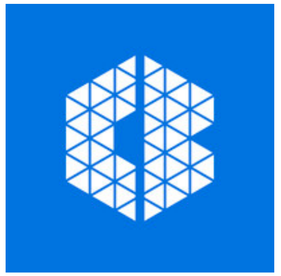
RIGHT?

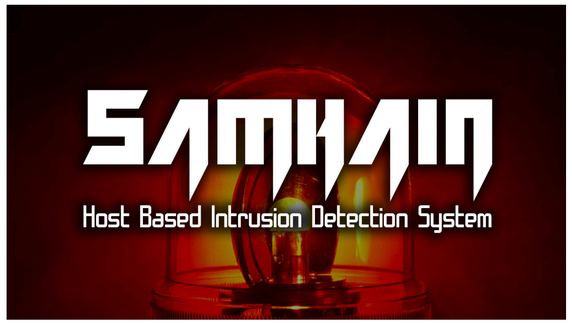
osquery

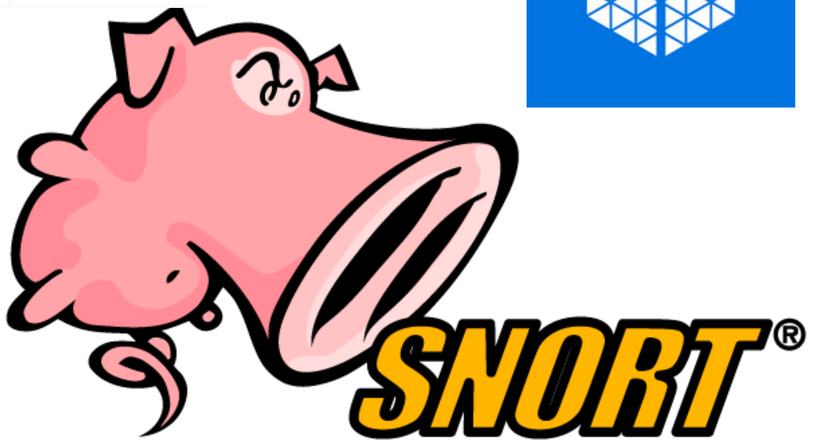
EnCase®

CROWDSTRIKE

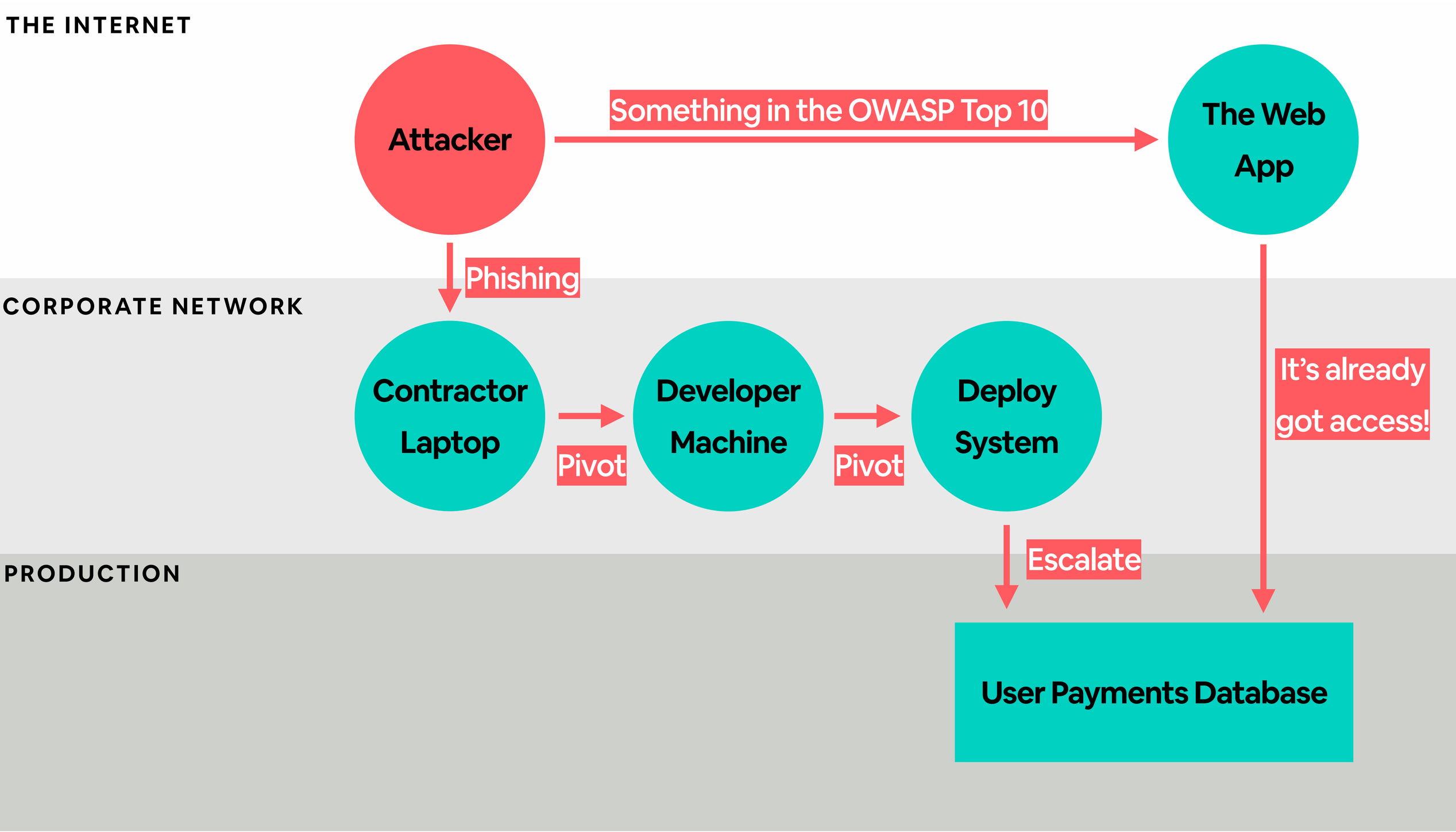
VOLATILITY







What about the web apps?


THE INTERNET
Attacker
Something in the OWASP Top 10
The Web App
Phishing
CORPORATE NETWORK
Contractor Laptop
Pivot
Developer Machine
Pivot
Deploy System
It's already got access!
Escalate
PRODUCTION
User Payments Database

Challenges

Missing Instrumentation

Shell histories, network logs, and default web server output won't tell you what you need to know.

Tons of Noise

Web vulnerability scanners are ubiquitous, so there's always attack noise on your public sites. Finding the exploits that worked can be hard.

Diverse Stacks

Applications vary a lot and the same tools may not work equally well across them.

An AppSec IR Framework

 **Track** the right metrics

 **Query** efficiently

 **Equip** the right tools

 **Plan** for the AppSec attack

FRAMEWORK ELEMENT 1

Track the Right Metrics

Metric Goals

AS A SET OF QUESTIONS

- “What are all the requests that user Alice has made?”
- “What are all the requests that Bob’s iPhone has made?”
- “How many requests look like they might be attacks?”

User & Device Tracking

- Enables anomaly tracking on a per-user or per-device basis
- Allows use of reputation systems on arbitrary traffic
- Establishes context to understand events

Identifying Single-Device Identity Changes

```
[nginx] t="2019-10-26T17:28:46+00:00" host=api.airbnb.com req="GET /v2/
messaging_syncs HTTP/1.1” user_id=5 device_id=1541994394_qkbYR7tUqv6nU8Se

[nginx] t="2019-10-26T17:28:52+00:00" host=api.airbnb.com req="GET /v2/
messaging_syncs HTTP/1.1” user_id=32211 device_id=1541994394_qkbYR7tUqv6nU8Se
```

Identifying Single-User Device Changes

```
[nginx] t="2019-10-26T10:53:23+00:00" host=api.airbnb.com req="GET /v2/
messaging_syncs HTTP/1.1” user_id=5 device_id=1541994394_qkbYR7tUqv6nU8Se

[nginx] t="2019-10-26T10:55:59+00:00" host=api.airbnb.com req="GET /v2/
messaging_syncs HTTP/1.1” user_id=5 device_id=1572074537_YTAwZWQhNUM3NWJi
```

Finding Malicious Requests

- Attack detection via signature matching is a hard problem, but that doesn't mean it's not useful
- Signal that indicates likely attacks can enrich other data sources

Evasion

A SIDENOTE

- Things like device tracking or web attack signature detection can be evaded
- They still provide you information in many cases
- They can force attackers to go slower and be less efficient

FRAMEWORK ELEMENT 2

Query Efficiently

Photo by Radek Grzybowski on Unsplash

WHAT DO WE DO WITH ALL THESE LOGS?

Data Science in IR

NO STATISTICS DEGREE REQUIRED

- Basic anomaly detection is not that hard and doesn’t require shelling out $2M to a dark-web-cyber-artificial-intelligence-machine-learning vendor
- Periodically do computations and surface results for really fast information lookups

Data Processing Strategy

- We use Apache Airflow to analyze our data in Hadoop/HDFS
- Periodic job:
  - Split requests by identity dimensions
  - Detect anomalies with simple statistics

Anomaly Detection Example

FINDING NEEDLES IN THE NEEDLE STACK

- Processing algorithm:
    1. Split global request data by (HTTP method, URL path, user ID)
    2. Analyze number of requests per user/method/path combo, get mean and standard deviation
    3. Identify users whose count of requests to a particular method/path are vastly different than the global population
- Good at finding exploitation of Insecure Direct Object References (IDORs) and some SQL injections

Vulnerability Refresher: the IDOR

“INSECURE DIRECT OBJECT REFERENCE”

Exploitation of a lack of access controls on an API.

```
GET /users_passport_photo/1

GET /users_passport_photo/2

GET /users_passport_photo/3
```

etc.

Vulnerability Refresher: Blind SQLi

“STRUCTURED QUERY LANGUAGE INJECTION”

Many SQL injection vulnerabilities only affect a single object, or require a lot of requests to extract information over a side channel.

```
GET /articles/
xyz'+UNION+SELECT+'a'+FROM+Users+WHERE+Username+=+'Administrator'+and+
SUBSTRING(Password, 1, 1)+>+'m'--
```

Forensic Examination vs. Proactive Detection

TWO BENEFITS OF GOOD QUERY CAPABILITIES

- Detected anomalies are always interesting after you know about a vulnerability
- With well-tuned anomaly detectors, you may be able to learn about currently-exploited vulnerabilities because of the unusual request patterns around them

FRAMEWORK ELEMENT 3

Equip the Right Tools

Querying All Those Metrics

PUTTING DATA IN EASY REACH DURING AN INCIDENT

- Think about the user experience in accessing the metrics you've gathered
- Our team primarily uses Jupyter notebooks, Superset, and Kibana

Rapid Blocking Tools

PREVENT ACCESS TO VULNERABLE CODE

- Don’t wait for your normal deploy process to block access to confirmed vulnerabilities
- Understand “feature flags” used by your engineers and learn how to find the switches for functionality
- Develop the capability to block features by request path / HTTP verb

Forced Client Upgrade Mechanisms

PUSH SAFE CODE TO USER DEVICES

- Some bugs require client changes for proper fixes. CSRF bugs are a good example.
- If you ship native apps to your users, you need to be able to get those users to update those apps.
- Android Inline Updates provide this natively; you'll need to implement your own solution for iOS.

FRAMEWORK ELEMENT 4

Plan for the AppSec Attack

Respond, don’t react.

Discovery

SOURCES OF INCIDENTS

- Bug bounty reports
- Penetration test findings
- “Weird logs”
- Seeing your data on Pastebin!
- Your own instrumentation, if tuned

Immediate Response

IDENTIFYING THE VULNERABILITY

- Easy if you started with a bug report
- Otherwise:
  - Anomaly tracking comes in handy
  - Talk to your experts
  - Find patterns in affected objects

Analysis & Postmortem

THE HARD QUESTIONS

- So you found a vulnerability that exposed data
  - Did anyone else find it?
  - Did anyone take anything?
  - Do you have breach notification obligations?

“

In the case of a personal data breach, the controller shall without undue delay and, where feasible, not later than 72 hours after having become aware of it, notify the personal data breach to the supervisory authority competent in accordance with Article 55, unless the personal data breach is unlikely to result in a risk to the rights and freedoms of natural persons.

Analysis & Postmortem

TIMELINE COMPRESSION

- Doing incident response *quickly* adds even more difficulty
- GDPR disclosure deadlines have led to some over-notification in the last year as security teams scramble to report impact

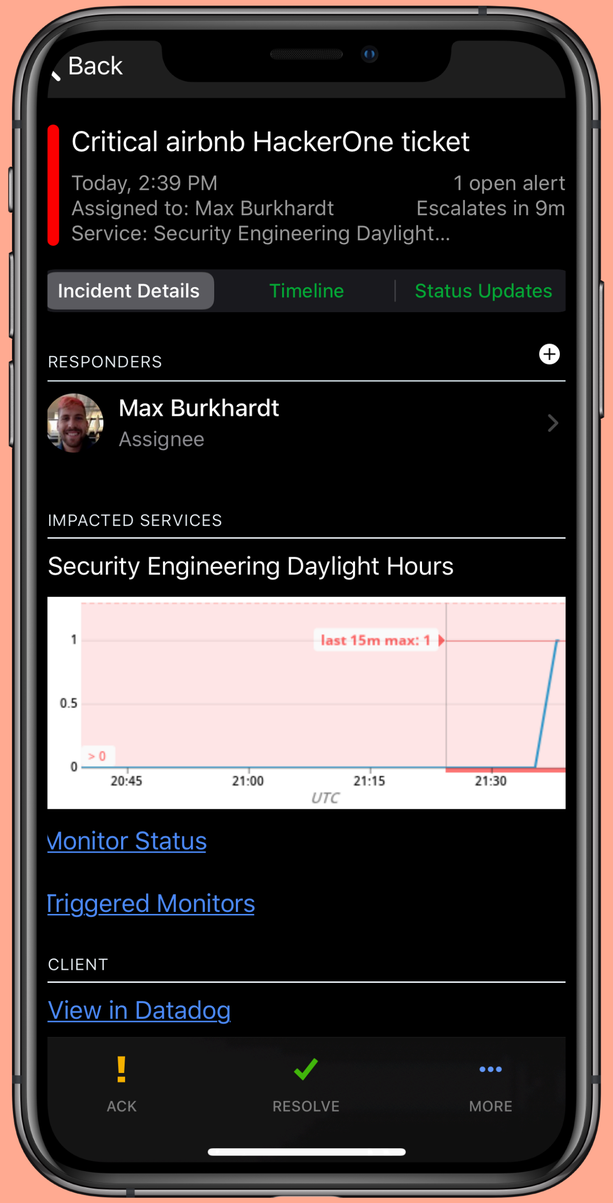
Analysis & Postmortem

USING THE TOOLS

- Anomaly detection has been key for us in following up on vulnerability investigations
- Log aggregation can be a huge time-sink after an investigation is underway — so try to do it beforehand

Real-world Example

USING THE FRAMEWORK AT AIRBNB



Setting the Stage

- We get a lot of good bugs via our bug bounty program
- This is the story of one of them
- Thanks to @rooting0x01 on HackerOne for this find!

Add an event

TITLE

AppSec Day

DATE & TIME

11/01/2019

Any time

LOCATION

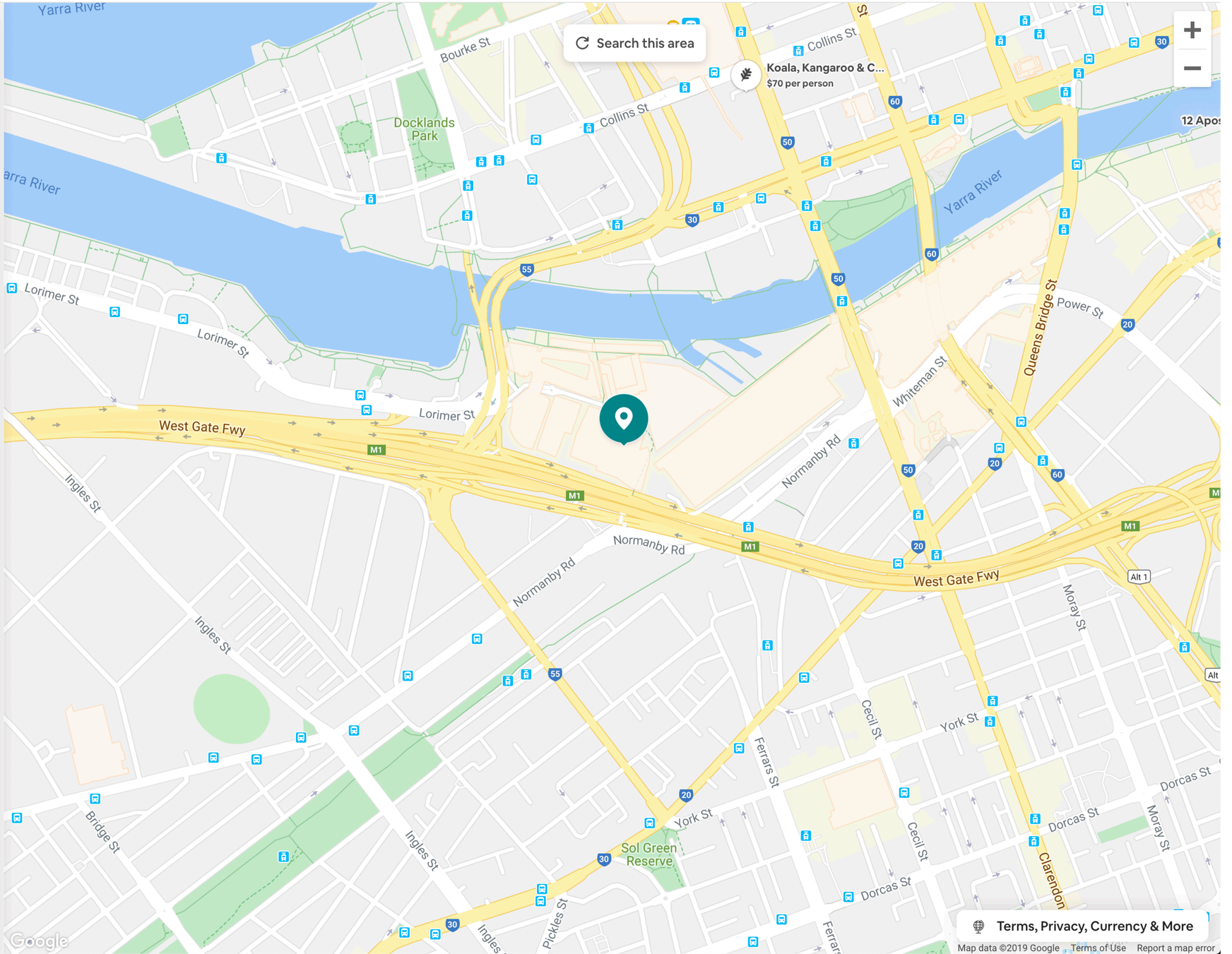
Goldfields Theatre, Convention Centre Place, South Wharf VIC, Australia

NOTES

Add more details

Invite guests

Add event



On the Wire

IT’S RESTFUL

```
GET /api/v2/event_guest_lists/FREEFORM_ENTRY/2064942
Host: www.airbnb.com
```

So of course...

INTEGER IDS STRIKE AGAIN

```
GET /api/v2/event_guest_lists/FREEFORM_ENTRY/2064943
Host: www.airbnb.com
```

What comes back?

USER DATA 😱


```
{
  "event_guest_list": {
    "guests": [
      {
        "can_manage": false,
        "email": "redacted@gmail.com",
        "id": "P_19990324",
        "name": "First Last",
        "schedulable_type": "FREEFORM_ENTRY",
        "status_key": "primary_booker",
        "user_id": 19990324,
        "name_short": "First",
        "label_single_character": "F"
      }
    ]
  }
}
```

Spring Into Action

ESCALATION

- We have PagerDuty connected to HackerOne via an Airflow job
- Tickets identified as major issues via human triage are routed to our team

Finding the Team

THE RIGHT FOLKS IN THE ROOM

- Two routes available:
  - Identify the service powering this endpoint → find owners
  - Escalate through product availability incident process

Parallelizing the Response

Dev Team:

Fix the bug.

Security Team:

Understand exploitation.

First Pass Query

```
SELECT * FROM core_security.user_id_url_path_anomalies WHERE
  url_path IN (
    '/api/v2/event_guest_lists/FREEFORM_ENTRY/NUMERIC_ID',
    '/v2/event_guest_lists/FREEFORM_ENTRY/NUMERIC_ID'
  ) AND
  method = 'GET' AND
  ds >= '2019-01-01' AND
  ds <= '2019-11-01'
```

AppSec - Endpoint Forensics Published

Edit dashboard

Forensics

Filters

Time range

Last month

url_path

× /api/v2/event_guest_lists/FREEFORM_ENTRY/NUMERIC

method

× GET

user_id

Endpoints of Interest

url_path	method	SUM(req_count)
/api/v2/event_guest_lists/FREEFORM_ENTRY/NUMERIC_ID	GET	177

UserID URL Path Anomalies

No data

What This Means

- We can really quickly identify if there appears to be large-scale exploitation of these types of issues
- This greatly changes how we engage with the process

Deeper Investigation

THE HOURS AND DAYS LATER

- Anomaly tables are fast to query because they condense info
- In-depth IR can take place over raw data tables in the time after triage

 **Track** the right metrics

 **Query** efficiently

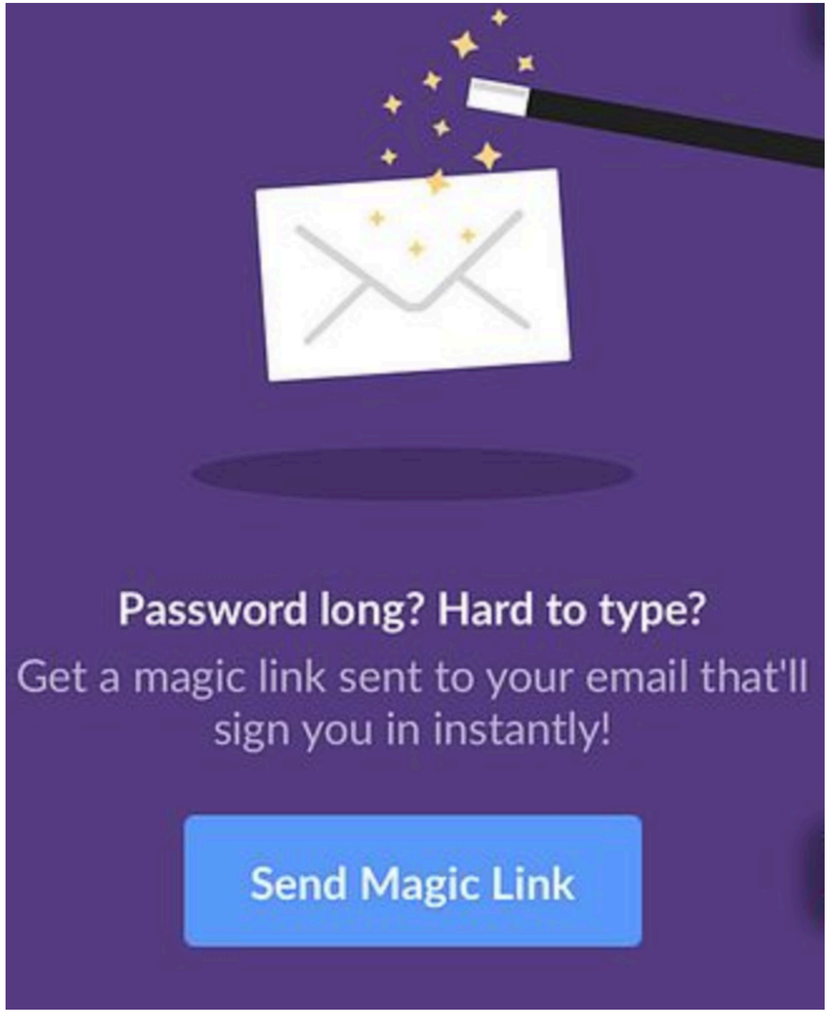
 **Equip** the right tools

 **Plan** for the AppSec attack

Defensive Tech I Love

Sites Without Passwords

- Use OAuth2 for social login plus email “magic links”
- Users clearly can’t manage passwords — why let them use them?
- A user losing control of their email means game over even if they have a password
- The new Webauthn standard is looking to be really cool too!



U2F Security Tokens

- A serious proposal to dealing with phishing
  - Uses web origin in its challenge-response protocol
  - Phishing sites can't "proxy" a challenge to steal someone's token
- Good example of easy-to-use crypto

Sandboxes by Default

- They’re everywhere
  - Mobile apps
  - Containerized software downloads
- Give control of devices back to the users who own them
- The face of malware on iOS and Android is fundamentally different than the “old days”

Your Infosec Career

There Are a Lot of Ways to Do This

LOTS OF PEOPLE HUNT SECURITY FLAWS

- Criminals
- Hacktivists
- Security researchers
- Pentesters
- Academics
- Defenders
- Governments

Can't recommend these

Playing Offense

PENTESTING, SECURITY RESEARCH, RED TEAM

- Builds a diverse skillset
- Really fun when your exploits land

Playing Defense

BLUE TEAM, SECURITY PRODUCT ENGINEERING

- You’re up against hard problems
- You get to eliminate threats and bug classes, one by one

Roles in Defense

SOFTWARE ENGINEER

- Write the tools that implement what you've learned here
  - Crypto toolkits
  - Frameworks to eliminate common bugs
  - Systems to analyze user activity for malicious indicators

Source: https://medium.com/airbnb-engineering/one-step-forward-in-data-protection-8071e2258d16

Roles in Defense

SECURITY ENGINEER

- Know the game — both sides of it
- Guide the development of software to mitigate security risk from the beginning

Roles in Defense

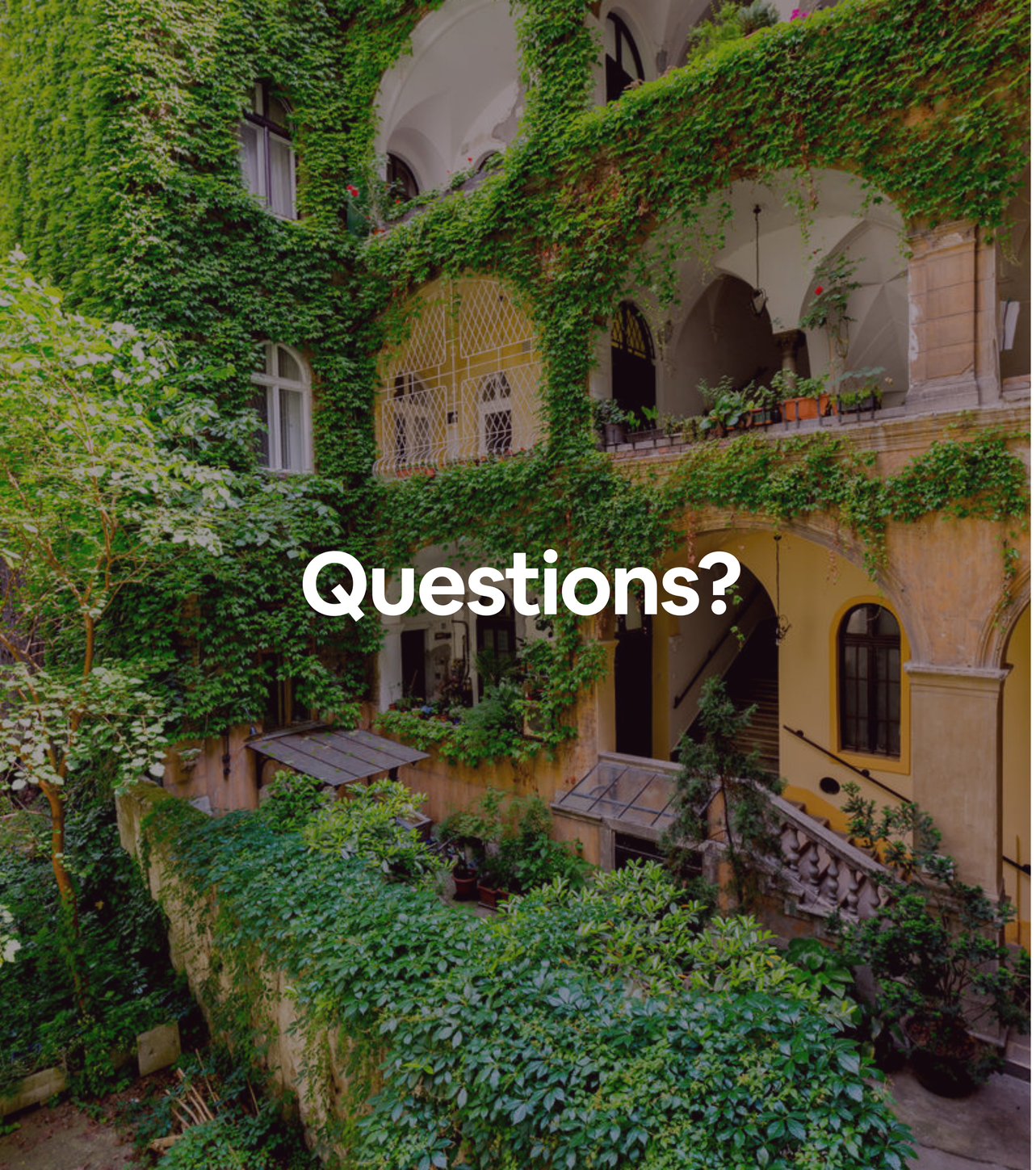
INTRUSION DETECTION ENGINEER

- Don’t give the adversary a moment’s rest during their attack
- Extract valuable signals, identity events, and respond with speed

Changing the Game

MAKE DEFENSE START WINNING

- Be more efficient with security effort
- Stop trying to scale defenses with people
- Turn security into an engineering problem



Questions?

Stay Connected

@maxb (Twitter)
root@maxb.fm