RSAConference2019
San Francisco | March 4–8 | Moscone Center

SESSION ID: SBX3-W2

First Steps in RF: Lessons Learned

Dave Weinstein
Engineering Manager
Android Security Assurance
Google

$ whoami

- Dave Weinstein

- Former AAA Game Developer (Networking Specialist)
- Security Engineer/Reverse Engineer/Analyst
- Engineering Manager for Android Security Assurance

Obviously, a background rich in networking and RF expertise...

Well, half right...

- Application development and network game development may use RF as a transport layer, but nothing I worked on was ever RF specific
- Mobile devices obviously are dependent on RF for communications, but a very small percentage of the code OR the security issues are in the RF layer itself

Actual RF Expertise (pre-2018)

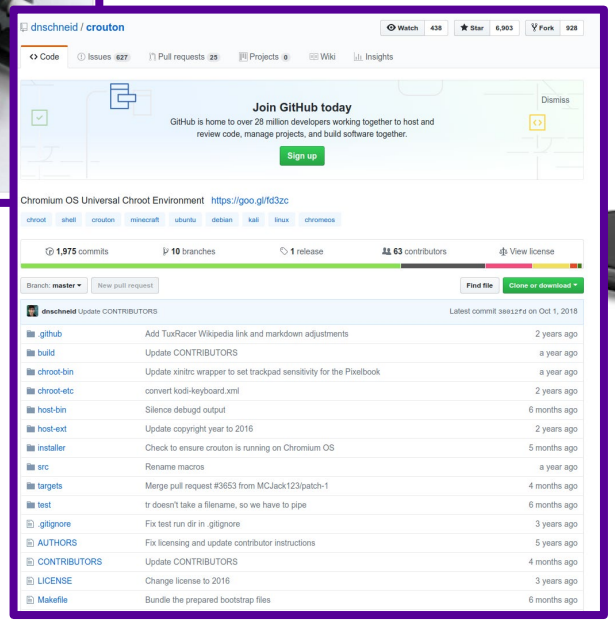
- Buying RF hardware
- Promising myself I would make time to learn how to use it
- Not actually learning to use it
- Buying better RF hardware
- Still not learning how to use it

I am (hopefully) the single least qualified person to talk about RF in this track!

So why are you even talking?

It’s all Rick’s fault...

DEF CON 2018 - Wireless Village



The tools and software I had tested worked…

…but I didn’t know enough to know what tools I’d need!

So I went ahead and tried anyway!

- Some things worked, but I wasn't skilled enough:
  - HackRF Portapack
- Some things just didn't work with my setup:
  - Blue Hydra
  - The nice new USB WiFi Interface I picked up on site
- Nonetheless, I was able to anchor a CTF Team!
  - The anchor is the heavy thing dragging everyone to the bottom, right? Close enough.

RSAConference2019

At this point you are probably wondering when the intro ends and the content will start

This talk is only tangentially about RF

Let’s consider a hypothetical Junior Engineer:

- Quiet in meetings
- Doesn’t ask many questions about design or codebase
- Spends a lot of time on Stack Overflow or searching the code base rather than asking for assistance

This is behavior we address in early career engineers...

- “It’s not a problem that it took longer, it’s a problem I didn’t know it was taking longer”
- “Don’t spend three days trying to answer a question that could be solved in an email”
- “Did you ask the team that wrote it if the feature you needed was on their roadmap before building a new one?”
- “Why didn’t you ask for help?”

And then we fall back into the same trap

- We're used to being experts, and being seen as experts
  - Status earned is something that is painful for most people to put at risk
- We are over-estimating our ability to learn on our own
- We expect that we can get competent enough fast enough to hide the up-front ignorance

Consciously modeling behavior

If we recognize that this is a pattern that we want to reinforce, we need to be explicit in demonstrating that.

Senior staff should be explicit and open about learning new things and starting from ignorance in those areas; it isn't enough to simply "not hide it".

Actual RF Expertise (after DEF CON 2018)

- SDR
  - Still don't know how to use them, but I do have a new one which is higher bandwidth and full duplex
- Bluetooth
  - Building Rust based libraries for working with the Ubertooth (or SDRs)
    - No, I don't really know Rust either. But this is a great project to learn it!
  - Looking to build tools for the Wireless CTF
    - I guess we'll find out if that worked out this week

RSA Conference2019

What we do matters more than what we say

Industry Antipatterns

“Work/life balance is a core value of our company.”

Also, we only promote the people who never take their vacation and are always working on weekends.

Lesson Learned: It is important to talk about work/life balance, so long as you don’t actually try to attain it.

Industry Antipatterns

“The work you are doing here is invaluable to the company, so I really need you to focus on that rather than taking on an additional project.”

We can’t promote you because you didn’t do anything new.

Lesson Learned: Mission critical operational roles are dead end jobs, and you’ll need to leave the company to advance.

Industry Antipatterns

“It’s important to ask questions, and to learn.”

But all the people in a position of authority or respect always make sure to never reveal their ignorance on anything.

Lesson Learned: Showing ignorance of anything is the mark of junior staff.

People aspire to respect and promotion, and will model their behaviour on people who achieve those.

Fundamental Issue: What we reward is what we encourage

Which of these should be directly rewarded at work?

A: Spending weekends learning a new programming language

B: Spending weekends skiing

C: Spending weekends as a maintainer on an Open Source project used at the company

D: Spending weekends coaching youth sports

E: Spending weekends learning a new technology or framework

If you aren’t paying for it in the first place, you shouldn’t be rewarding it.

Direct and indirect rewards

If learning a new programming language is a requirement for a given role, then you should be budgeting **paid** time for employees to master it.

If someone makes a case that a language should be a requirement, and that the team or company should adopt it, then you reward them for the **paid** work in setting up the workflow and training flow for everyone to move to that language.

Direct and indirect rewards

If maintaining an Open Source project is something the company wants to reward, the company should budget the **paid** time and cost for that work as part of the normal business expectations.

Direct and indirect rewards

As with programming languages, if expertise with a given technology becomes useful, they should be rewarded for the **paid** work that they apply that expertise to, not for how they acquired it.

If you expect people to master a technology as part of their job, you should **pay** for the training and ramp-up time.

Direct and indirect rewards

If you create a culture in which doing off-hours, unpaid work becomes a requirement for success, you are inherently creating a culture which is **hostile** to people with other interests or other obligations.

Principles

- Senior staff need to openly model the behavior you expect of more junior staff
- The behavior we reward is by definition the behavior that we have chosen to encourage
- If you aren't willing to pay for it in the first place, you shouldn't reward working on it

Principles into Action

Over the next 3 days

- Look for opportunities to demonstrate the behavior you want to see exhibited by the rest of your team

Over the next 30 days

- Look for misalignments between behaviors you reward, behaviors you want, and behaviors you are willing to pay for

Over the next 3 months

- Bring your team goals and rewards into alignment

RSAConference2019

Discussion