If You Can’t Beat ‘Em Join ‘Em

Tips For Running a Successful Bug Bounty Program

Grant McCracken
Daniel Trauner

Bay area cyber-security meetup
April 21, 2016
Grant

- Technical Account Manager @Bugcrowd
  - formerly an ASE
- Before that, Whitehat
- Did some traveling
- Music
Dan

- AppSec Engineer (ASE) @Bugcrowd
- Before that, Fortify SCA @HPSR
  - Static analysis -- lots of languages
  - Focus on iOS
- Art History/Collecting!
Bug Bounty Programs
wut
A (Brief) History of Bug Bounty Programs
Why?
Do you really want to let people attack you?

Source: http://hyperboleandahalf.blogspot.com/2010_06_01_archive.html
Yes! (They’re doing it anyways…)

CLEAN ALL THE THINGS!

Source: http://hyperboleandahalf.blogspot.com/2010_06_01_archive.html
You vs. and Them

TEAMWORK
Always works.

fakeposters.com
Who are these people?

- All over the place!
  - All ages
  - All levels of experience
  - All over the world
  - Users and non-users

- Passionate about security
Value

- Lots of eyes
- Only pay for valid results
- Shows a more advanced security posture
- Better overall reputation!
CASH MONEY
I’m gonna make it rain.
“Touch the code, pay the bug.”
Pre-Launch
Scope, scope, scope

● Step 0...
  ○ Basic resources/requirements to run a program

● The researcher’s universe
  ○ Leave nothing open to interpretation
  ○ Understand your attack surface
  ○ The path of least resistance
Focus

- You might care about specific:
  - Targets
  - Vuln types
  - Functionalities (e.g. payment processing)

- How?
  - Incentives
  - Create a focused program
Exclusions

- You might not care about:
  - (Low-impact) “Low-hanging fruit”
  - Intended functionality
  - Known issues
  - Accepted Risks
  - Issues resulting from pivoting
Access

● Prod vs. Staging
  ■ Make sure it can stand up to testing!
    1. Scanners/contact forms
    2. Pentesting requests

● Access
  ○ Easier = better
    ■ IoT? iOS? Special?
    ■ What will researchers need?
      1. cc/ssn/etc
    ■ Researcher environments
What a shared environment looks like...
Remember…
Post-Launch
Expectations, expectations, expectations...

Bay area cyber-security meetup 2016 – If You Can’t Beat ‘Em Join ‘Em

© Bugcrowd 2016
Expectations, expectations, expectations...
Communication is Key

● Researchers like:
  ○ Concise, unambiguous responses
    ■ ESL
  ○ Quick responses
  ○ Predictable time to reward

● Stay on top of these issues!

● Public disclosure?
Define a VRT

- **Vulnerability Rating Taxonomy**
- **For you:**
  - Speed up triage process
  - Track your organization’s security posture
- **For them (if published):**
  - Focus on high-value bugs (& avoid wontfix)
  - Feel a sense of trust (goes with brief)
- **Establish a “Vulnerability Roundtable”**
Meanwhile, IRL...
Recipe for disaster:

1. Don’t provide known issues
2. Don’t consider exclusions
3. Sneaky brief changes
### Instructure

<table>
<thead>
<tr>
<th></th>
<th>2013 (Pentest)</th>
<th>2014 (Bug Bounty)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>High</td>
<td>1</td>
<td>25</td>
</tr>
<tr>
<td>Medium</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>Low</td>
<td>2</td>
<td>16</td>
</tr>
</tbody>
</table>

Source: [https://www.canvaslms.com/security](https://www.canvaslms.com/security)
It's super effective!