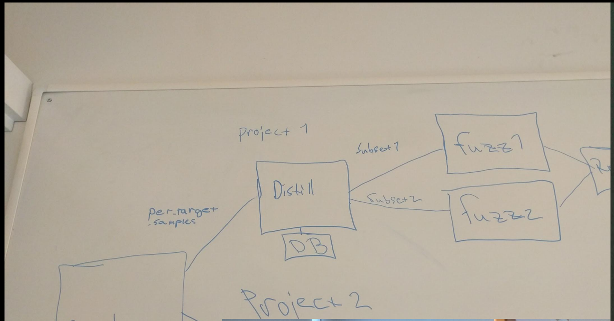
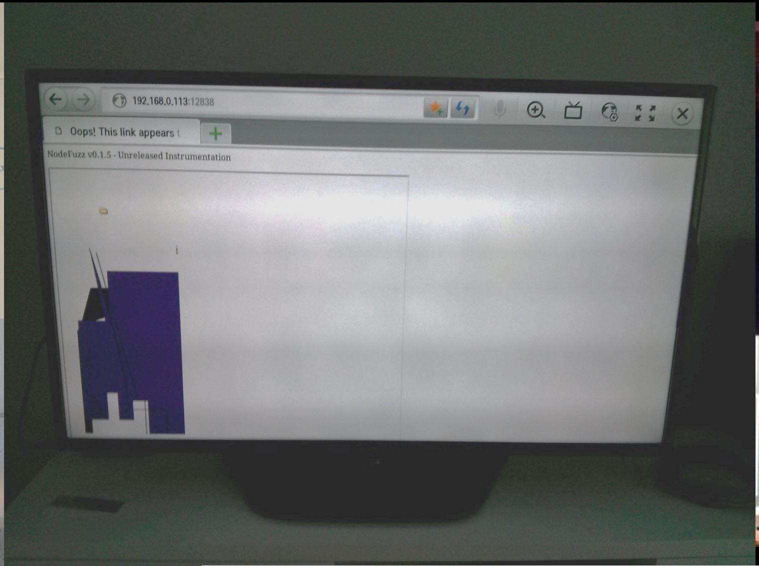
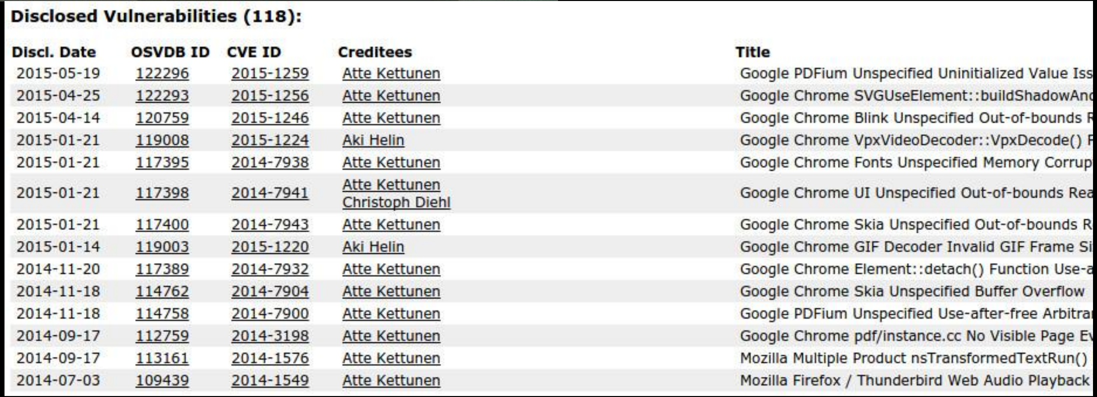
Do's and Don'ts:
A Day Of Browser Bug Bounty Hunting

Atte Kettunen
@attekett
OUSPG




192.168.0.113:12838
Oops! This link appears t
NodeFuzz v0.1.5 - Unreleased Instrumentation



Disclosed Vulnerabilities (118):

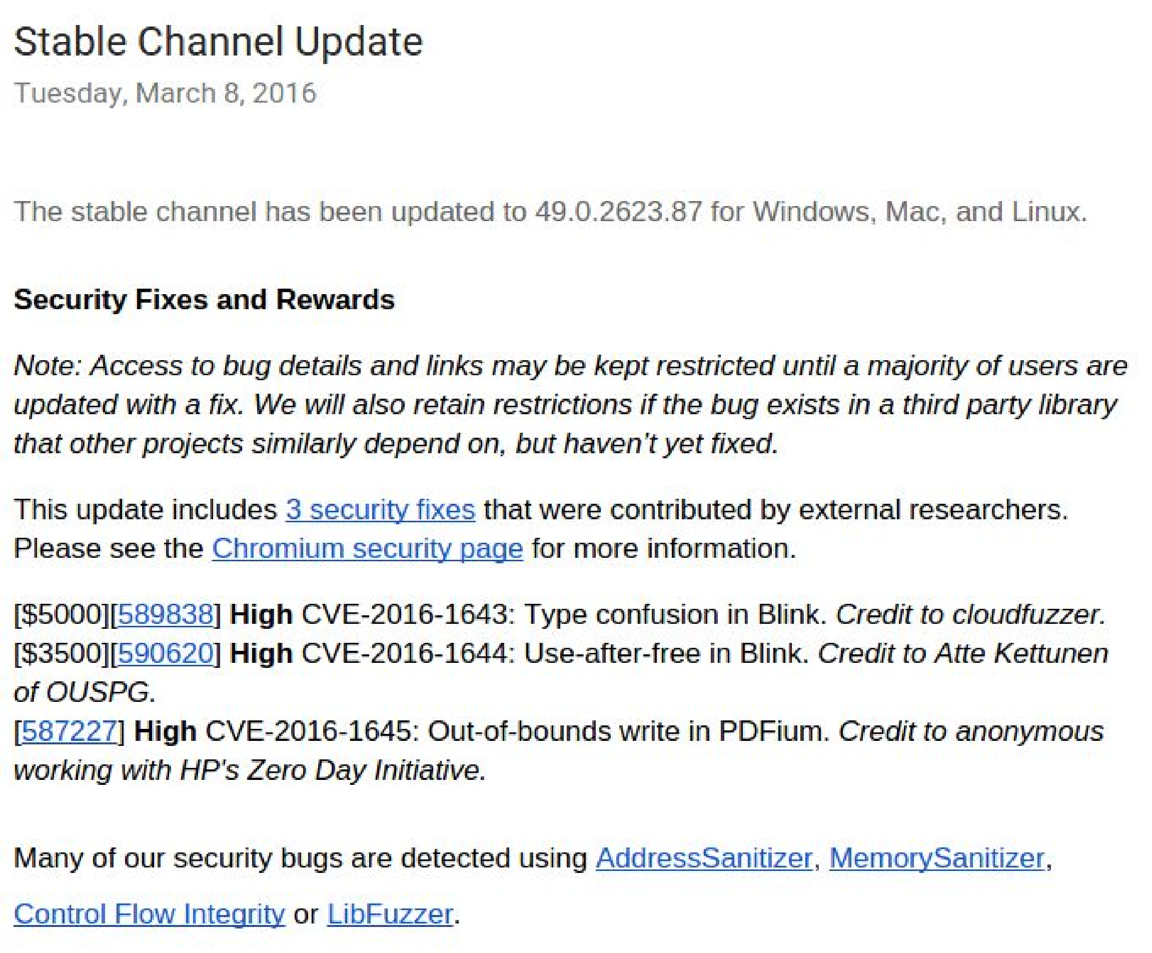
Discl. Date	OSVDB ID	CVE ID	Creditees	Title
2015-05-19	122296	2015-1259	Atte Kettunen	Google PDFium Unspecified Uninitialized Value Iss
2015-04-25	122293	2015-1256	Atte Kettunen	Google Chrome SVGUseElement::buildShadowAnd
2015-04-14	120759	2015-1246	Atte Kettunen	Google Chrome Blink Unspecified Out-of-bounds R
2015-01-21	119008	2015-1224	Aki Helin	Google Chrome VpxVideoDecoder::VpxDecode() F
2015-01-21	117395	2014-7938	Atte Kettunen	Google Chrome Fonts Unspecified Memory Corrup
2015-01-21	117398	2014-7941	Atte Kettunen Christoph Diehl	Google Chrome UI Unspecified Out-of-bounds Rea
2015-01-21	117400	2014-7943	Atte Kettunen	Google Chrome Skia Unspecified Out-of-bounds R
2015-01-14	119003	2015-1220	Aki Helin	Google Chrome GIF Decoder Invalid GIF Frame Si
2014-11-20	117389	2014-7932	Atte Kettunen	Google Chrome Element::detach() Function Use-a
2014-11-18	114762	2014-7904	Atte Kettunen	Google Chrome Skia Unspecified Buffer Overflow
2014-11-18	114758	2014-7900	Atte Kettunen	Google PDFium Unspecified Use-after-free Arbitrar
2014-09-17	112759	2014-3198	Atte Kettunen	Google Chrome pdf/instance.cc No Visible Page Ev
2014-09-17	113161	2014-1576	Atte Kettunen	Mozilla Multiple Product nsTransformedTextRun()
2014-07-03	109439	2014-1549	Atte Kettunen	Mozilla Firefox / Thunderbird Web Audio Playback

Bounty Programs

- Vendor offered program, which researchers can use to report vulnerabilities directly to the vendor and get a **monetary** compensation.
- IMHO: Hall-of-fame mentions and swag are not bounties.

Bounty Programs

- Flow of vulnerability information and research data between researcher community and the software vendor



Stable Channel Update

Tuesday, March 8, 2016

The stable channel has been updated to 49.0.2623.87 for Windows, Mac, and Linux.

Security Fixes and Rewards

Note: Access to bug details and links may be kept restricted until a majority of users are updated with a fix. We will also retain restrictions if the bug exists in a third party library that other projects similarly depend on, but haven't yet fixed.

This update includes 3 security fixes that were contributed by external researchers. Please see the Chromium security page for more information.

[$5000][589838] **High** CVE-2016-1643: Type confusion in Blink. *Credit to cloudfuzzer.*
[$3500][590620] **High** CVE-2016-1644: Use-after-free in Blink. *Credit to Atte Kettunen of OUSPG.*
[587227] **High** CVE-2016-1645: Out-of-bounds write in PDFium. *Credit to anonymous working with HP's Zero Day Initiative.*

Many of our security bugs are detected using AddressSanitizer, MemorySanitizer,

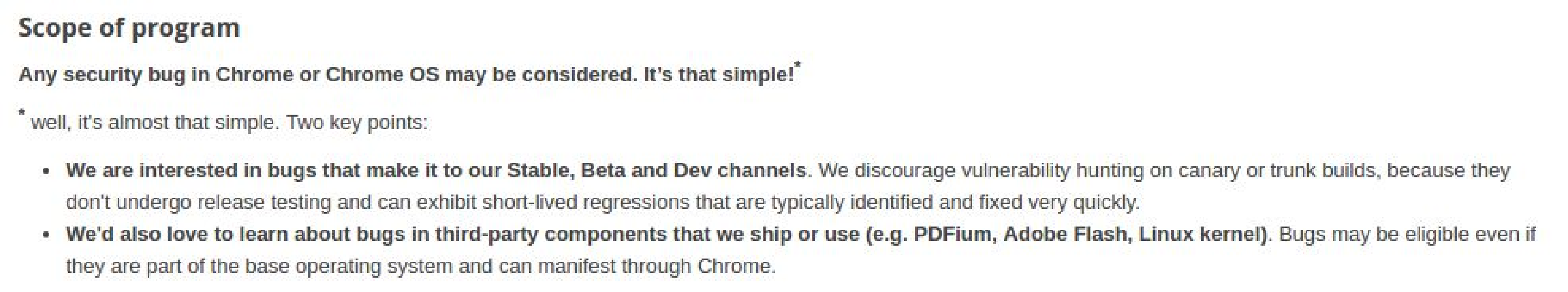
Control Flow Integrity or LibFuzzer.

TL;DR

- Bounty Program Rules
    - Scope
        - What, in where?
    - Qualifying Vulnerabilities
        - What kind of bugs?
    - Rewards
        - Baseline?
        - Bonus?

Rules - Scope

- Platform
- Versions
- Plugins/Addons/Extensions/Modules
- Attack vectors



Scope of program

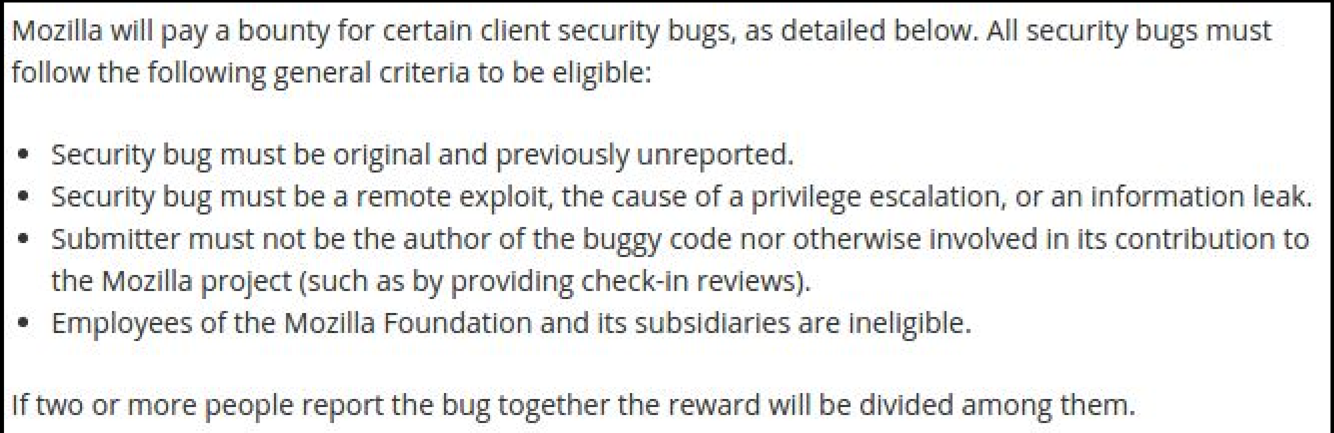
Any security bug in Chrome or Chrome OS may be considered. It's that simple!*

* well, it's almost that simple. Two key points:

- **We are interested in bugs that make it to our Stable, Beta and Dev channels**. We discourage vulnerability hunting on canary or trunk builds, because they don't undergo release testing and can exhibit short-lived regressions that are typically identified and fixed very quickly.
- **We'd also love to learn about bugs in third-party components that we ship or use (e.g. PDFium, Adobe Flash, Linux kernel)**. Bugs may be eligible even if they are part of the base operating system and can manifest through Chrome.

Qualifying Vulnerabilities

- Good: Severity
- Better: Vulnerability types
  - RCE, DoS, clickjack, account theft etc…



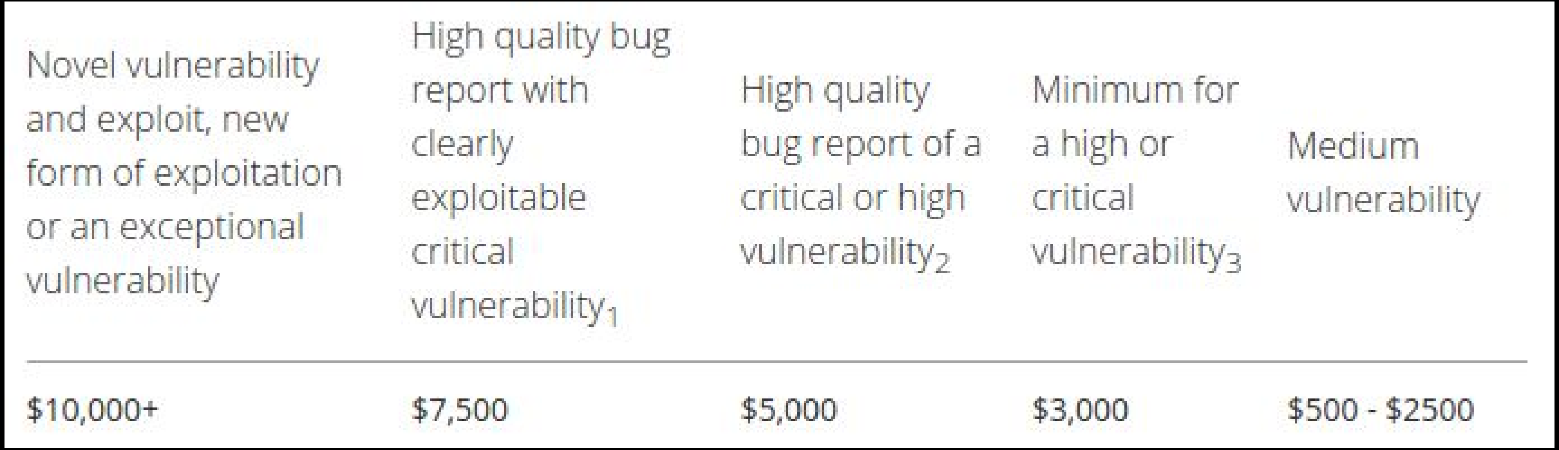
Mozilla will pay a bounty for certain client security bugs, as detailed below. All security bugs must follow the following general criteria to be eligible:

- Security bug must be original and previously unreported.
- Security bug must be a remote exploit, the cause of a privilege escalation, or an information leak.
- Submitter must not be the author of the buggy code nor otherwise involved in its contribution to the Mozilla project (such as by providing check-in reviews).
- Employees of the Mozilla Foundation and its subsidiaries are ineligible.

If two or more people report the bug together the reward will be divided among them.

Rewards

Baseline:



Novel vulnerability and exploit, new form of exploitation or an exceptional vulnerability	High quality bug report with clearly exploitable critical vulnerability[1]	High quality bug report of a critical or high vulnerability[2]	Minimum for a high or critical vulnerability[3]	Medium vulnerability
$10,000+	$7,500	$5,000	$3,000	$500 - $2500

Rewards

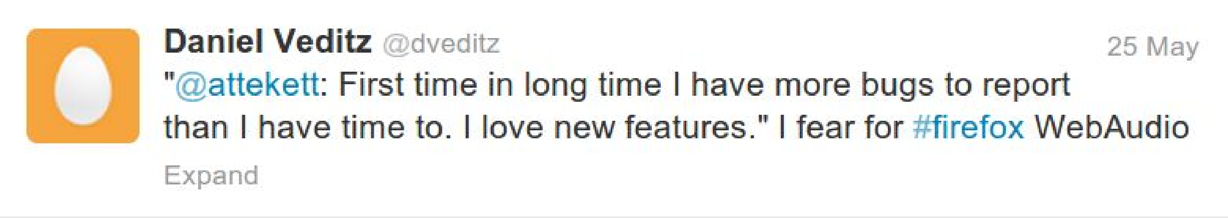
- Higher rewards for special bugs and/or feats.
  - Google:
    - Patch reward: +$500-1,337$
    - Proof of exploitability: +$1,000
    - Impact to a significantly wider range of products than just Chromium: +$1,000
    - “Cool bugs”: +$n

Finding targets

- READ THE RULES!
- Release notes
- Vendor blogs
- Twitter/Facebook/Google+
- IRC
- Changelogs

Finding targets - WebAudio

"The API has been designed to allow *modular routing*.(UAF) Basic audio operations are performed by **audio nodes** that are linked together to form an *audio routing graphs*.(UAF/BOF) Inside a same context, several sources are supported, with different kind of channel layout.(UAF/BOF) This modular design allows for great flexibility and for the creation of complex audio functions and of dynamic effects.(BOF)" - MDN



Daniel Veditz @dveditz 25 May

"@attekett: First time in long time I have more bugs to report than I have time to. I love new features." I fear for #firefox WebAudio

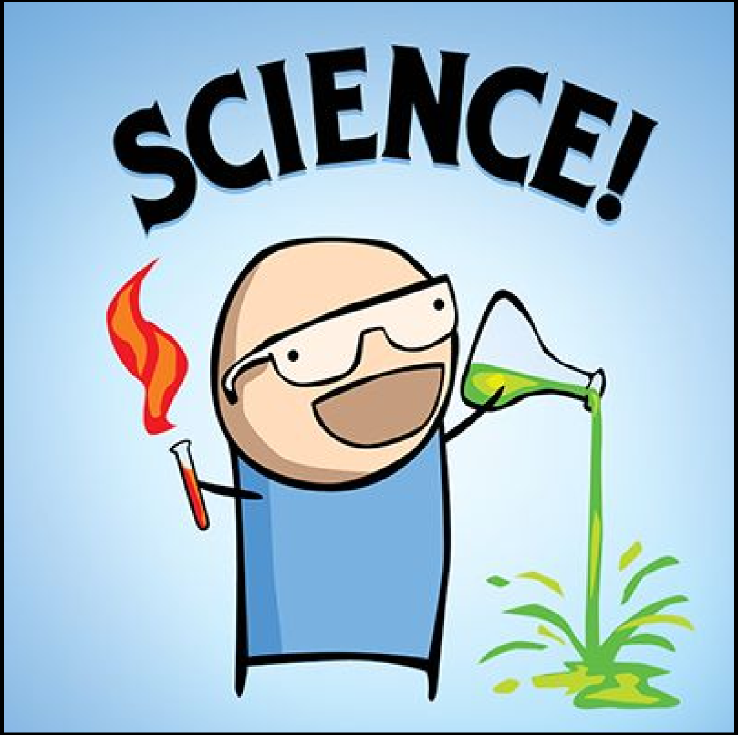
Expand

Finding targets - WebAudio

Bugs found:

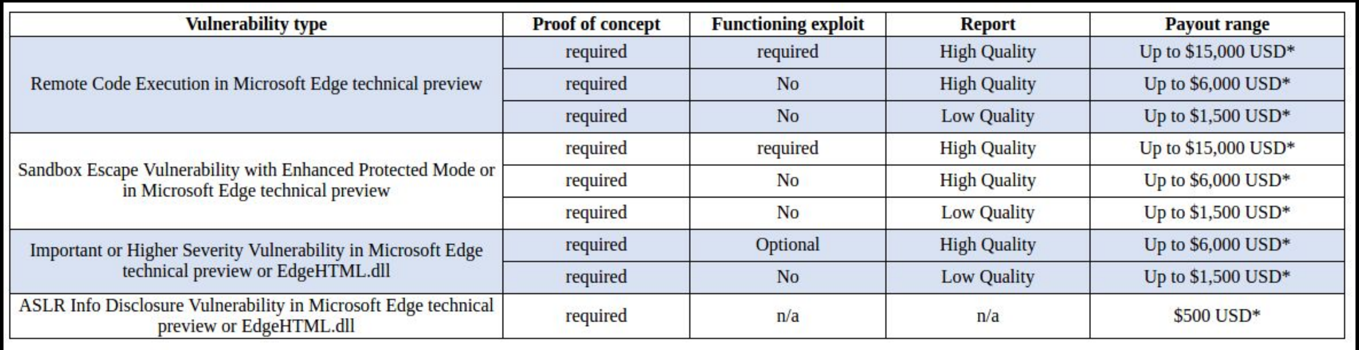
- Chrome - 4 UAF, 3 BOF
- Firefox - 1 UAF, 8 BOF

Writing testing tools



Reporting

- Good: Good reports should, and often do do get higher reward.
- Bad: Rules are often ambiguous about “good” and “bad”



Vulnerability type	Proof of concept	Functioning exploit	Report	Payout range
Remote Code Execution in Microsoft Edge technical preview	required	required	High Quality	Up to $15,000 USD*
	required	No	High Quality	Up to $6,000 USD*
	required	No	Low Quality	Up to $1,500 USD*
Sandbox Escape Vulnerability with Enhanced Protected Mode or in Microsoft Edge technical preview	required	required	High Quality	Up to $15,000 USD*
	required	No	High Quality	Up to $6,000 USD*
	required	No	Low Quality	Up to $1,500 USD*
Important or Higher Severity Vulnerability in Microsoft Edge technical preview or EdgeHTML.dll	required	Optional	High Quality	Up to $6,000 USD*
	required	No	Low Quality	Up to $1,500 USD*
ASLR Info Disclosure Vulnerability in Microsoft Edge technical preview or EdgeHTML.dll	required	n/a	n/a	$500 USD*

Reporting

Often used ambiguous terms:

- “Proof of Concept”
- “Exploitable”
- “High quality bug report”
- “Impact (critical|severe|high|medium|low|.*)”

Reporting

- Bare minimum:
  - Application version
  - Platform info
  - DETAILED description how to reproduce the vulnerability
  - (Description of the vulnerability, the security impact and the attack scenario)
- Program rules sometimes define additional requirements.

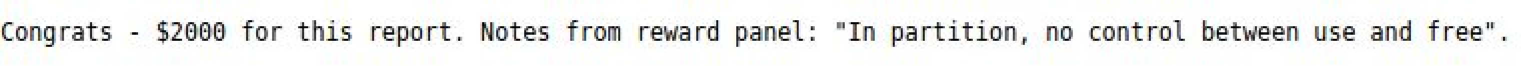
Reporting

- IMHO:
  - 15min for $2k reward, 3wk for $3k / $7.5k???
  - Is proving exploitability, or exploit development, worth the time?

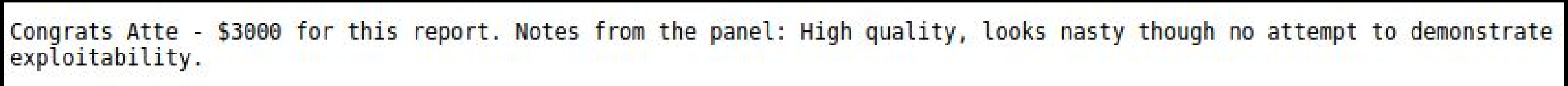
- I use that time to develop better bug finding tools.

Communication

- Interaction creates value for both sides!
- Creates trust
- IMHO: 90% of the experience



Congrats - $2000 for this report. Notes from reward panel: "In partition, no control between use and free".



Congrats Atte - $3000 for this report. Notes from the panel: High quality, looks nasty though no attempt to demonstrate exploitability.

Reporting

- Vendor side people are as interested to have a working bounty program as the researchers
- Ask before wasting your time.

Third-party vulnerability programs

- “The Zero Day Initiative (ZDI), founded by TippingPoint, is a program for rewarding security researchers for responsibly disclosing vulnerabilities.”
- “ZERODIUM is a premium exploit acquisition platform for high-end zero-days and advanced vulnerability research.”
- “SecuriTeam Secure Disclosure (SSD) provides the support you need to turn your experience uncovering security vulnerabilities into a highly paid career.”

Q&A

Thank you!