

Bug bounties

experience from both sides



© Warner Bros. Entertainment Inc. (s18)

@davwwwx

PoC || GTFO







@davwwwx

hackerone



INTIGRITI
ETHICAL HACKING PLATFORM



YES WE H~~A~~C^K



HackerOne

Vulnerability disclosure should be safe, transparent, and rewarding.

<https://hackerone.com> · [@Hacker0x01](#)

Reports resolved: 501 | Assets in scope: 11 | Average bounty: \$500

[Submit report](#)

Bug Bounty Program
Launched on Nov 2013

Managed by HackerOne

Includes retesting (?)

Bounty splitting enabled (?)

Policy Hacktivity Thanks Updates (0) Collaborators

Rewards

Low Medium High Critical

<https://hackerone.com>

Response Efficiency

21 hrs

Average time to first response

<https://hackerone.com/security?type=team>

@davwwwx



Krisp

Krisp is an AI-powered app that removes background noise and echo from meetings leaving only human voice.

<https://krisp.ai> · [@krispHQ](#)

Reports resolved
2

Assets in scope
14

Average bounty
\$500-\$1k

[Submit report](#)

[Edit Page](#)

Bug Bounty Program
Launched on Jul 2021

Managed by HackerOne

Includes retesting (?)

[Bookmark](#) [Subscribe](#)

[Policy](#) [Hacktivity](#) [Thanks](#) [Updates \(0\)](#)

Rewards

Low

Medium

High

Critical

\$100

\$500

\$1,000

\$5,000

\$100

\$250

\$750

\$1,500

Response Efficiency

2 days

Average time to first response

3 days

Average time to triage

<https://hackerone.com/krisp?type=team>

@davwwwx

Submit Vulnerability Report



You're about to submit a report to Krisp. Provide as much information as possible about the potential issue you have discovered. The more information you provide, the quicker Krisp will be able to validate the issue. If you haven't yet, please remember to review our [Security Page](#).

1

Asset

Select the attack surface of this issue.



Select Asset Type...

<https://krisp.ai>

Domain • Critical • Eligible for bounty

<https://account.krisp.ai>

Domain • Critical • Eligible for bounty

<https://krisp.ai/security/#responsible-disclosure>

@davwwwx



U.S. Dept Of Defense

<https://bit.ly/2Rt5QhM> · @DC3VDP

[Submit report](#)

Reports resolved

17190

Assets in scope

Vulnerability Disclosure
Program

Launched on Nov 2016

[Policy](#) [Hacktivity](#) [Thanks](#) [Updates \(0\)](#)

Policy

DoD Vulnerability Disclosure Policy

Purpose

This expanded program is intended to give security researchers terms and conditions for conducting vulnerability discovery activities directed at publicly accessible Department of Defense (DoD) information systems¹, including web properties, and submitting discovered vulnerabilities to DoD. If questions arise, please take no action until that action is discussed with the VDP lead at the Department of Defense Cyber Crime Center (DC3).

Overview

Response Efficiency

6 hrs

Average time to first response

6 hrs

Average time to triage

● 100% of reports

Meet [response standards](#)

Based on last 90 days

<https://hackerone.com/deptofdefense?type=team>

@davwwwx



@davwwwx

HackerOne external programs



Twilio

<http://twilio.com> · @twilio

[Contact Security Team](#)

External Program

Policy

<https://docs.hackerone.com/hackers/create-a-directory-page.html>



security.txt

```
# Our security address  
Contact: mailto:security@example.com  
  
# Our PGP key  
Encryption: https://example.com/pgp-key.txt  
  
# Our security policy  
Policy: https://example.com/security-policy.html
```

<https://example.com/.well-known/security.txt>



FIREBOUNTY

YES WE H~~F~~CK

HOME

ADD VDP

CREATE YOUR BUG BOUNTY

ABOUT

GET VDP FINDER EXTENSION



THE RIGHT PATH TO COORDINATED VULNERABILITY DISCLOSURE.

ZERODISCLO

YES WE H~~F~~CK

Name ▾

Search a program

Search

« Previous

1

2

3

...

402

Next »

20100 policies found (out of 20100)

NAME

REWARDS ▾

SCOPE TYPES ▾

TYPE ▾

CREATED AT ▾

UPDATED AT



LE GROUPE LA POSTE VDP

Le Groupe La Poste Vulnerability Disclosure Policy The safety and security of our customers' d...

THANKS

GIFT

HOF

REWARD

n/a

Bug bounty

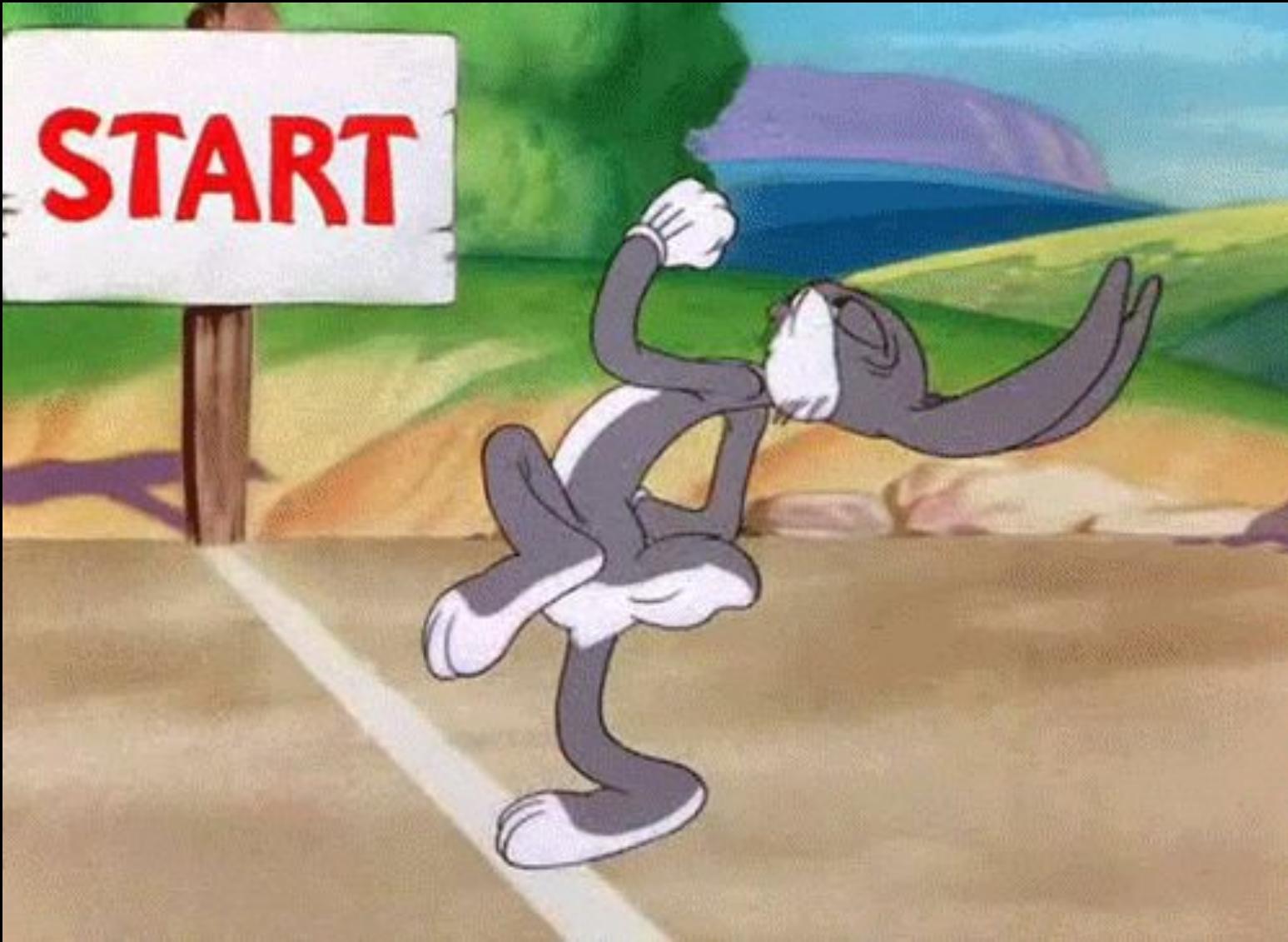
2021-08-19

2021-08-20

<https://firebounty.com/>

@davwwwx

Live Hacking Events



@davwwwx



<https://www.hackerone.com/ethical-hacker/hack-hard-have-fun-increase-security>

 2 derision	 1 jonathanbo...	 3 zseano						
1077 REPUTATION	44 REPORTS	49 BOUNTIES	1151 REPUTATION	29 REPORTS	59 BOUNTIES	346 REPUTATION	9 REPORTS	26 BOUNTIES

REPUTATION REPORTS BOUNTIES

REPUTATION REPORTS BOUNTIES

REPUTATION REPORTS BOUNTIES

4.  cache-money	349	9	17
5.  spaceraccoon	664	23	45
6.  intidc	193	4	9
7.  the_arch_angel	216	15	45

Note: leaderboard is sorted by total bounty amount earned

View the complete results!

FINISHED
\$832,135 bounty paid

 ANNOUNCEMENT:

Hacking, hanging with our pets and sharing our favorite travel memories!



<https://www.hackerone.com/ethical-hacker/hack-hard-have-fun-increase-security>

@davwwwx



ArmBounty

BOUNTIES



BOUNTIES EVERYWHERE



HOW TO GET STARTED IN BUG BOUNTY (9x PRO TIPS)

473K views • 2 years ago



STÖK

So here are the tips/pointers I give to anyone that's new to Bug bounty / bounties and apptesting. 1. Sign up for Hackerone to get ...

HOW DO YOU GET STARTED?

11:22



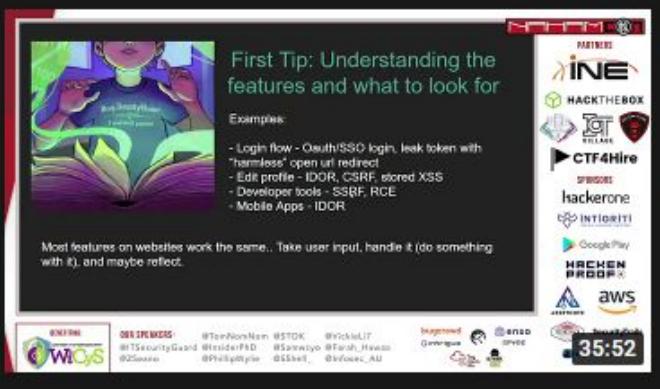
The Truth About Bug Bounties

74K views • 1 year ago



The Cyber Mentor

*We are a participant in the Amazon Services LLC Associates Program, an affiliate advertising program designed to provide a ...



NahamCon2021 - Putting Your Mind to It: Bug Bounties for 12 Months - @zseano

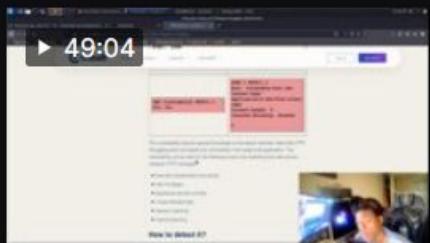
11K views • 4 months ago



Nahamsec

NahamCon2021 - Hosted by TheCyberMentor, John Hammond and NahamSec Sponsors ----- WiCys ...

Past videos



Trying to find \$20,000 bug bounties

Kingsharpe

War Zone

6 views · 19 hours ago



Hacks & Chips - Chill Dev/AMA...work on Homelab, bug bounties...who knows....!socials !root !commands

Cyber_Insecurity ✅

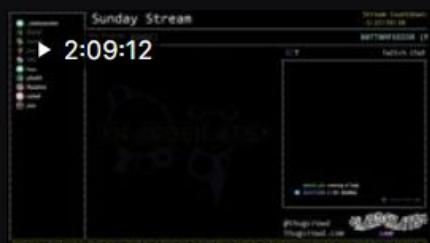
Science & Technology

1.7K views · 2 months ago

Desktop Development

Software Development

AMA



Episode 17: Bug Bounties

hardchat

158 views · 3 years ago



Episode 17.5: Bug Bounties Part 2

hardchat

57 views · 3 years ago



Highlight: Continuing Bug Fables Chapter 6 & Bounties || !discord

SheenvAmava

Intro to Bug Bounty Hunting and Web Application Hacking

Insiders guide to ethical hacking and bug bounty hunting with Ben Sadeghipour (@NahamSec)

4.6 ★★★★★ (988 ratings) 10,979 students

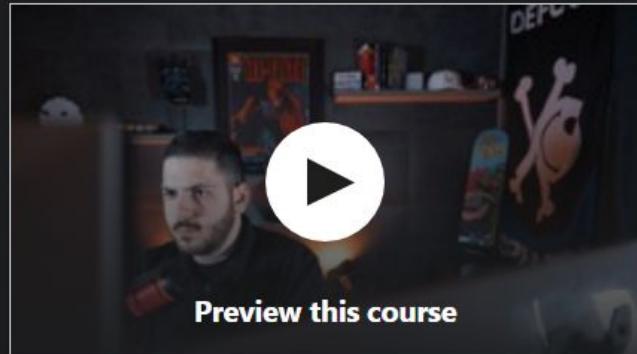
Created by [Ben Sadeghipour](#)

Last updated 2/2021 English English [Auto]

[Wishlist](#) ❤

[Share](#) ➔

[Gift this course](#)



\$13.99 \$89.99 84% off

⌚ 5 hours left at this price!

[Add to cart](#)

[Buy now](#)

30-Day Money-Back Guarantee

This course includes:

- ▶ 5 hours on-demand video
- ♾ Full lifetime access
- 📱 Access on mobile and TV
- 🖨 Certificate of completion

What you'll learn

- ✓ Learn 10+ different vulnerability types
- ✓ Basics of Reconnaissance
- ✓ Understand how bug bounties work
- ✓ Includes practical hands on labs to practice your skills
- ✓ Ability to exploit basic web application vulnerabilities
- ✓ How to approach a target
- ✓ Write better bug bounty reports
- ✓ Hack Websites for Ethical Hacking



Bug**Bounty**Hunter



TEST YOUR SKILLS ▾



USEFUL RESOURCES ▾



OUR MEMBERS & EVENTS ▾



MEMBERS AREA

Helping you connect the **bug** to **bounty**

We aim to become your go to place for everything bug bounties. Learn how to test for security vulnerabilities on web applications with our various real-life web applications and gain the confidence to begin applying your newly found knowledge on bug bounty programs. Browse and digest security researcher tutorials, guides, writeups and let us help you on your journey.

Made with ❤ by @zseano
Artwork by laracallejaillustrations

<https://www.bugbountyhunter.com/>

@davwwwx

#bugbounty

2.2M views

...



Weakness



Bounty

\$20,000

Severity

Critical (9 ~ 10)

Participants



Visibility

Disclosed (Full)



kyle.tobener

Remote Code Executio...

Steam Wallet Bug



Tamper traffic
with BURP



kyle.tobener

I LOVE this find. SO cle...

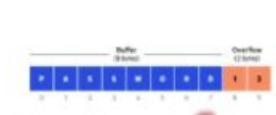
(Basically)

\$25k Snap RCE



kyle.tobener

This was a tough vuln...



Buffer overflow example

What is a buffer overw...

Attack by overwriting...

Attacking by overwriting...

Pwn2own buffer

kyle.tobener



ashwinonfire

Bug Bounty Part 1



kyle.tobener

Razer Mouse

@davwwwx

Recon subdomains and gau to search vuls DalFox

- Explaining command

```
assetfinder testphp.vulnweb.com | gau | dalfox pipe
```

Recon subdomains and Screenshot to URL using gowitness

- Explaining command

```
assetfinder -subs-only army.mil | httpx -silent -timeout 50 | xargs -I@ sh -c 'gowitness single @"'
```

Extract urls to source code comments

- Explaining command

```
cat urls1 | html-tool comments | grep -oE '\b(https?|http)://[-A-Za-z0-9+&@#/%?=~_|!:,.;]*[-A-Za-z0-9+&@#/%=~_
```



Patrik Fehrenbach 🐚 @ITSecurityguard · 21h
Yay, I was awarded a \$10,000 bounty on @Hacker0x01!
hackerone.com/patrik #TogetherWeHitHarder

Usually I don't **yay** but this time, I had to

Merry Christmas 🎁🎄



HackerOne profile - patrik
- <http://www.it-securityguard.com>
🔗 [hackerone.com](http://hackerone.com/patrik)

🗨 10

⬇ 4

❤ 208



...



Akshay Kerkar @AkshayKerkar13 · Dec 21
Yay, I was awarded a @Sony ❤️ swag for reporting a vulnerability.
#bugbounty #togetherwehitharder #hackerone



🗨 3

⬇ 1

❤ 24



Daniel @danielabs · Oct 24, 2017
Thanks @ncsc_nl for the t-shirt #bugbounty #swag



🗨 5

⬇ 18

❤ 35



...



Jenish Sojitra @_jensec · 20h
Yay, I was awarded a \$6,000 bounty on @Hacker0x01 for a CORS!

To make a CORS critical you can scrape account info of victim's accounts via this tool with requests relay. It will use value from response.
[bugpoc.com/testers/other/...](http://bugpoc.com/testers/other/)

hackerone.com/jensec #TogetherWeHitHarder



HackerOne profile - jensec
Security & Finance -
🔗 [hackerone.com](http://hackerone.com/jensec)

🗨 13

⬇ 70

❤ 389



@davwwwx



Grammarly

Grammarly makes sure everything you type is clear, effective, and mistake-free.

[Submit report](#)

<https://www.grammarly.com>

Reports resolved

166

Assets in scope

14

Average bounty

\$200-\$400

Bug Bounty Program
Launched on Dec 2018

Managed by HackerOne

Includes retesting (?)

[Policy](#) [Hacktivity](#) [Thanks](#) [Updates \(8\)](#)

Rewards

Low

Medium

High

Critical

Capture the Flag

-

-

-

\$100,000

\$500

\$2,500

\$12,500

\$25,000

Response Efficiency

2 days

Average time to first response

3 days

Average time to triage

5 months

Average time to resolution

<https://hackerone.com/grammarly?type=team>

@davwwwx

**YOU GET IN FOR
FREE, YOU GET IN FOR FREE**



FREE ENTRY FOR EVERYONE!!!

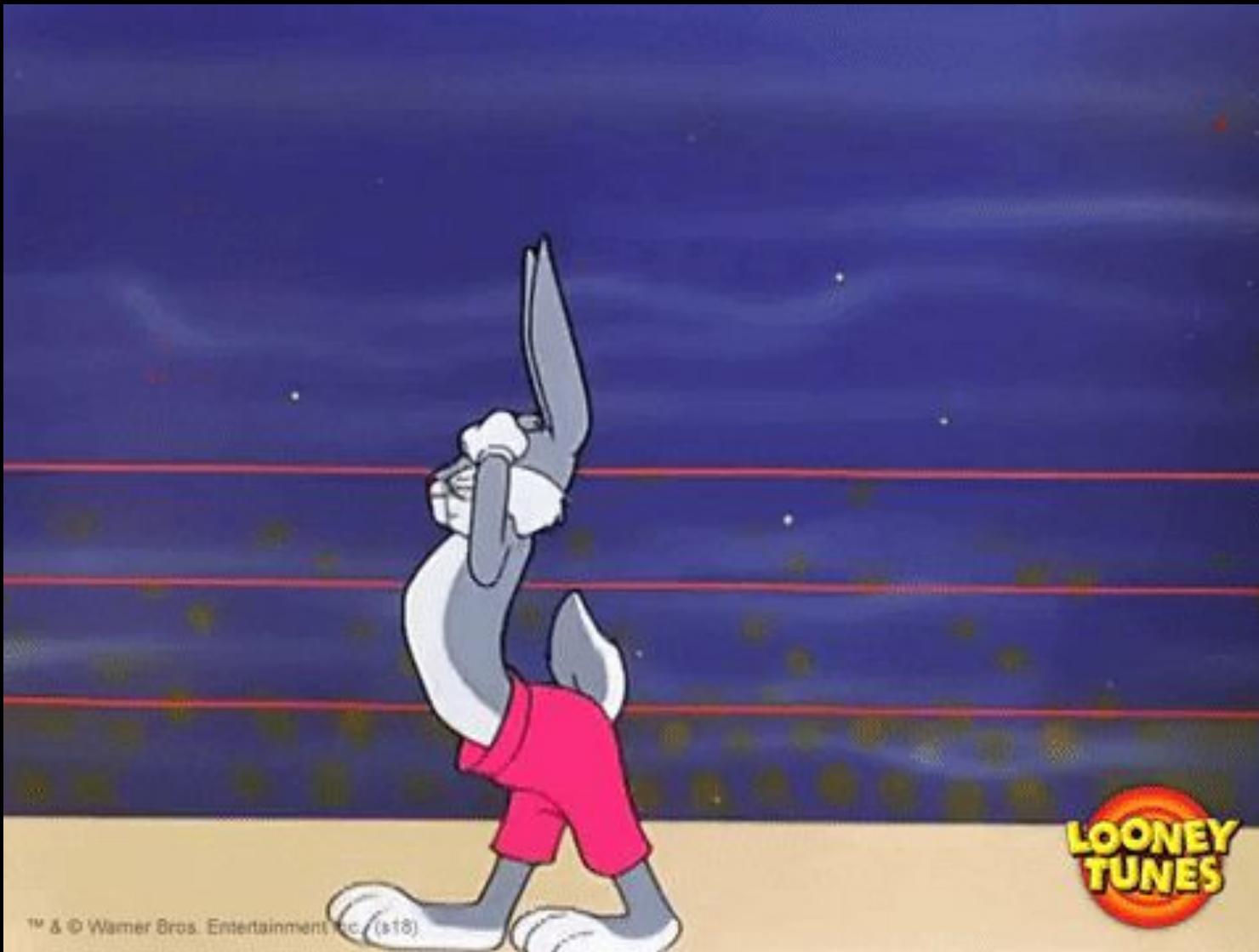


© & © Warner Bros. Entertainment Inc. (517)

@davwwwx

Program	Launch date	Reports resolved	Bounties minimum	Bounties average
 Node.js Bounty splitting	03 / 2018	37	\$500	\$250

<https://twitter.com/ghidraninja/status/1432668917412028420>



@davwwwx

Rank	Hacker	Reputation	Impact	Signal
1	 Eric todayisnew	112826	16.57	5.49
2	 Mr Hack try_to_hack	53338	17.21	5.27
3	 Sergey Markov sergeym	37802	15.37	5.27
4	 d0xing d0xing	30053	19.90	6.67
5	 Sean Melia meals	28512	19.41	5.80
6	 Black Ashes nullelite	26435	16.13	3.78

<http://bugbounty.space/>

@davwwwx



@davwwwx



УЧИТЬСЯ УЧИТЬСЯ И ЕЩЕ РАЗ УЧИТЬСЯ!



EAT



SLEEP



WORK



WALK

Team rating

2021 2020 2019 2018 2017 2016 2015 2014 2013 2012
2011

Place	Team	Country	Rating
1	perfect blue	🇺🇸	965.598
2	DiceGang	🇺🇸	747.557
3	More Smoked Leet Chicken	🇷🇺	732.093
4	Bushwhackers	🇷🇺	713.743
5	Super Guesser		705.564
6	Plaid Parliament of Pwning	🇺🇸	694.231
7	Never Stop Exploiting	🇨🇳	551.385
8	C4T BuT S4D	🇷🇺	537.756
9	organizers		524.657
10	Katzebin	🇨🇳	517.758

[Full rating](#) | [Rating formula](#)

Past events

With scoreboard

All

WMCTF 2021

Aug. 30, 2021 01:00 UTC | On-line | [Weight voting in progress](#)

Place	Team	Country	Points
1	Ph0t1n1a	🇨🇳	50.000
2	Nu1L	🇨🇳	34.806
3	r3kapig	🇨🇳	26.448

192 teams total | [Tasks and writeups](#)

YauzaCTF 2021

Aug. 29, 2021 12:00 UTC | On-line | [Weight voting in progress](#)

Place	Team	Country	Points *
1	SPRAVEDLIVAR RUSH A	🇷🇺	0.000
2	fargate	🇷🇺	0.000
3	Bulba Hackers	🇬🇧	0.000

227 teams total | [Tasks and writeups](#)

Learning materials and labs

Latest

**OAuth
authentication**

6 labs

**HTTP Host header
attacks**

6 labs

**Business logic
vulnerabilities**

11 labs

**Web cache
poisoning**

13 labs

Featured

SQL injection

**Cross-site scripting
(XSS)**

**Cross-site request
forgery (CSRF)**

XXE injection

PentesterLab will help you get to the next level!

Category:

All Categories ▾



CVE-2021-37xxx
Code Review Badge



PHP Snippet #01
Code Review Badge



PHP Snippet #02
Code Review Badge



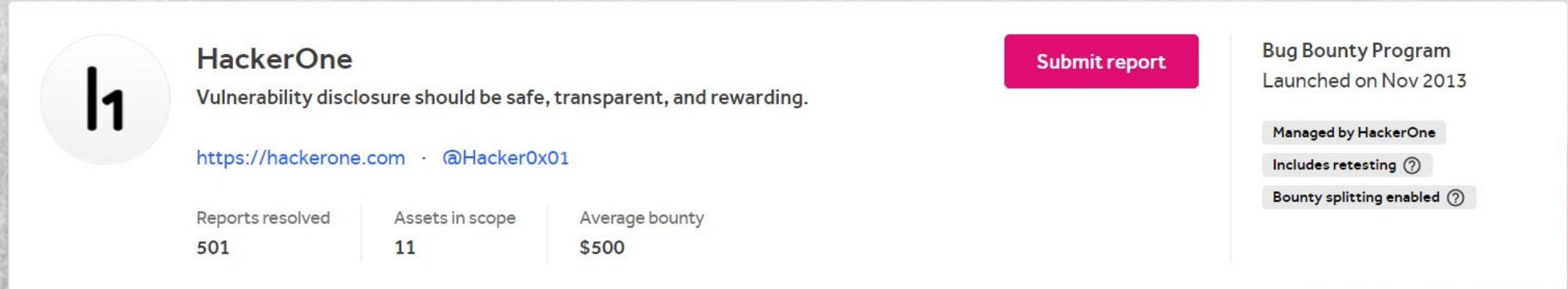
PHP Snippet #03
Code Review Badge

<https://pentesterlab.com/exercises>

@davwwwx

Some Terminology

Severity



HackerOne
Vulnerability disclosure should be safe, transparent, and rewarding.

<https://hackerone.com> · [@Hacker0x01](#)

Reports resolved: 501 | Assets in scope: 11 | Average bounty: \$500

[Submit report](#)

Bug Bounty Program
Launched on Nov 2013

Managed by HackerOne
Includes retesting (?)
Bounty splitting enabled (?)

Policy Hacktivity Thanks Updates (0) Collaborators

Rewards

Low

Medium

High

Critical

<https://hackerone.com>

Response Efficiency

21 hrs

Average time to first response

<https://hackerone.com/security?type=team>

@davwwwx



Public

Open

intigriti/intigriti/Detail

Description

At intigriti, we practice what we preach. We've built the platform with the greatest care and attention for security, but all software contains bugs and we are no exception to this rule. We encourage you to responsibly disclose any security vulnerabilities they may encounter and will reward you accordingly.

Bounties

Low	Medium	High	Critical	Exceptional
Tier 2 € 150	€ 750	€ 2,500	€ 5,000	€ 7,500

[*View changes](#)

Detail

Leaderboard

All aboard!

Please log in or sign up on the platform

For obvious reasons we can only allow submissions or applications for our program with a valid intigriti account.

It will only take 2 minutes to create a new one or even less to log in with an existing account, so don't hesitate and let's get started. We would be thrilled to have you as part of our community.

[Log in or sign up](#)

<https://app.intigriti.com/programs/intigriti/intigriti/detail>

@davwwwx

CVSS

CVSS v3.0 Calculator ?

Critical 10.0

Attack Vector ?

Network Adjacent Local Physical

Attack Complexity ?

Low High

Privileges Required ?

None Low High

User Interaction ?

None Required

Scope ?

Unchanged Changed

Confidentiality ?

None Low High

Integrity ?

None Low High

Availability ?

None Low High

<https://www.first.org/cvss/v3.0/user-guide>

<https://www.first.org/cvss/v3.0/examples>

<https://kb.intigriti.com/en/articles/5041991-intigriti-s-contextual-cvss-standard>

Weakness

CWE

Weakness

Select the type of the potential issue you have discovered. Can't pick just one? Select the best match or submit a separate report for each distinct weakness.



Select Weakness Type... ▾

Incorrectly Specified Destination in a Communication Channel (CWE-941)

Use of Incorrectly-Resolved Name or Reference (CWE-706)

Improper Neutralization of Special Elements (CWE-138)

Missing Critical Step in Authentication (CWE-304)

XSS Using MIME Type Mismatch (CAPEC-209)

XML Injection (CWE-871)

<https://cwe.mitre.org/>

@davwwwx

States

NEW

TRIAGED

RESOLVED

TRIAGE

PENDING

ACCEPTED

RESOLVED

NEEDS MORE INFO

ACCEPTED RISK

INFORMATIVE

RETESTING

DUPLICATE

OUT OF SCOPE

NOT APPLICABLE

SPAM

<https://docs.hackerone.com/hackers/report-states.html>

<https://kb.intigriti.com/en/articles/3379382-submission-lifecycle>



Triage



@davwwwx



davwwwx created the submission
[REDACTED]

Unread messages



voljin [triage]
[REDACTED]

Hi davwwwx,

Thank you for the great written report and all the time you are making to look into the security of our new and fresh program!

We reviewed your report and were able to reproduce it. Therefore, we are going to forward your submission towards the security team of [REDACTED]. They will take a deeper look into this and come back at you as soon as possible!

Keep up the good work and have a lovely weekend,

Voljin



voljin changed the severity from **Exceptional** to **Exceptional** (10.0)
[REDACTED]

@davwwwx



[company]

Thank you davwwwx for reporting this one to us. As part of the program details, please do not communicate about this vulnerability, or even one found in our program. We're on it, and will solve this as soon as possible.

Kristof



[company] changed the status from **Pending** to **Accepted**



[company] rewarded a bounty of [REDACTED] to **davwwwx**



davwwwx [researcher]

How to write a report

Description*

What is the vulnerability? In clear steps, how do you reproduce it?

Please replace all the [square] sections below with the pertinent details. Remember, the more detail you provide, the easier it is for us to triage and respond quickly, so be sure to take your time filling out the report!

Summary:

[add summary of the vulnerability]

Steps To Reproduce:

[add details for how we can reproduce the issue]

1. [add step]

1. [add step]

1. [add step]

Supporting Material/References:

[list any additional material (e.g. screenshots, logs, etc.)]

* [attachment / reference]

TIMELINE



haxta4ok00 submitted a report to [HackerOne](#).

Mar 23rd (5 months ago)

Summary:

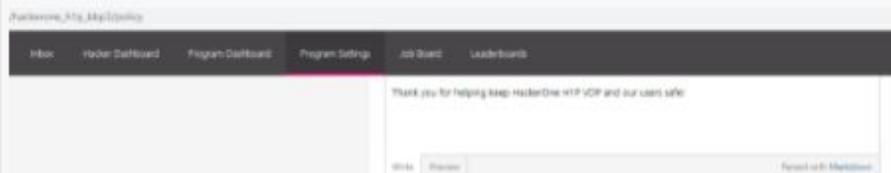
Hi team,

Our team noticed that you(program) can attach files to the policy page. These files can be anything, images, text, archive, etc.In other words, these files may or may not contain sensitive information. Our team believes that the data that can be attached in different vectors is high . Therefore, in the CVSS calculator, we set Confidentiality: **High** .

Also, the HackerOne platform slightly confuses customers in this situation. When the client tries to delete a file from the tab where the file is attached, the page shows that the file was deleted, and after clicking the "Update policy page" button, it shows that it was successfully updated. But the page does not reload, and the client sees that the file was indeed deleted. We also tested this on the endpoint, and indeed. The update takes place without the involvement of the Attachment file. But after you refresh the policy edit page, this file will appear again. But visually, the client initially believes that the file was deleted, until he refreshes the page and sees it. We believe this is misleading to the customer

Image F1239141: file_1.png 20.48 KiB

[Zoom in](#) [Zoom out](#) [Copy](#) [Download](#)



<https://hackerone.com/reports/1132606>

@davwwwx

IMPACT > EVERYTHING



TM & © Warner Bros. Entertainment Inc. (s17)

@davwwwx

What to not report

Blindly pasting automatic scanner reports

BUG REPORT 1:

Issue: SSL cookie without secure flag set

Severity: Medium

Confidence: Certain

Host: <https://krispai.zendesk.com>

Path: /embeddable/config

REQUEST:

GET /embeddable/config HTTP/1.1

Host: krispai.zendesk.com

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:78.0) Gecko/20100101 Firefox/78.0

Accept: */*

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Origin: <https://krisp.ai>

Connection: close

Referer: <https://krisp.ai/security/>

Best practices

Missing DNS records

13

#410245

Missing Certificate Authority Authorization rule

Share:

TIMELINE



171217 submitted a report to [HackerOne](#).

Sep 16th (3 years ago)

Certificate Authority Authorization (supported by LetsEncrypt and other CAs) allows a domain owner to specify which Certificate Authorities should be allowed to issue certificates for the domain. All CAA-compliant certificate authorities should refuse to issue a certificate unless they are the CA of record for the target site. This helps reduce the threat of a bad guy tricking a Certificate Authority into issuing a phony certificate for your site.

The CAA rule is stored as a DNS resource record of type 257. You can view a domain's CAA rule using a DNS lookup service:

<https://dns.google.com/query?name=hacker101.com&type=257&dnssec=true>

<https://dns.google.com/query?name=ctf.hacker101.com&type=257&dnssec=true>

hacker101 should set a CAA record to help prevent misissuance of a certificate for its domains.

Reference Report : <https://hackerone.com/reports/129992>

Impact

Misissuance of a certificate

Reported September 16, 2018 10:34am +0400



171217

Participants



State Resolved ()

Reported to [HackerOne](#)

Disclosed April 11, 2019 10:29pm +0400

Severity None (0.0)

Weakness None

CVE ID None

Account de... None

CAA RECORD
MISSING

NOT APPLICABLE



paragonie-scott Paragon Initiative Enterprises staff closed the report and changed the status to ● Spam.

Nov 3rd (5 years ago)

paragonie-scott Paragon Initiative Enterprises staff requested to disclose this report.

Nov 3rd (5 years ago)

paragonie-scott Paragon Initiative Enterprises staff disclosed this report.

Nov 3rd (5 years ago)

[hackerone_hero](#) posted a comment.

Any reason ? why did you close this as spam ?

How can you close without giving any reason?

Don't you see , detailed report with image explanation of vulnerability ?

paragonie-scott Paragon Initiative Enterprises staff posted a comment.

Nov 4th (5 years ago)

1. Your report is way out of scope. For the purposes of our Hacker One program, we don't care one bit if paragonie.com gets owned up, we only care that people who use our open source software don't.

2. You plagiarized an dmrc.org article instead of linking to it.

3. You indicated that this is a "very critical" bug and checked off the following categories:

- Authentication
 - Command Injection
 - Cross-Site Request Forgery (CSRF)
 - Cross-Site Scripting (XSS)

 hackerone_hero

Participants



State ● Spam ()

Reported to **Paragon Initiative Enterprises**

Disclosed November 3, 2016 9:30am +0400

Severity Critical (9 ~ 10)

Weakness *None*

CVE ID *None*

Account de... *None*

TIMELINE



fabiothebest89 submitted a report to [Skyliner](#).

Sep 16th (5 years ago)

First of all I will start with some theory.

DNS is a system to translate a domain name to an ip address. Normally a computer automatically trust a DNS server and connects to the IP provided by the DNS server. This is prone to security issues because a malicious wifi network, an attacker on your router, a compromised ISP, or any other man-in-the-middle attack can redirect a DNS request to a server of their choice (a fake one) and this can allow phishing, malware spreading, botnets and can also cause Denial of Service in a way because the original server will become unreachable unless you know its IP address and you use it for establishing a TCP connection. An attacker can do domain hijacking, create forged DNS updates, unauthorised zone transfers, cache poisoning and DoS.

How can we solve all these problems? The answer is: implementing DNSSEC.

And you did it, but you did it wrong.

If you want to read about the DNSSEC standard check RFC4034(<https://www.ietf.org/rfc/rfc4034.txt>) and RFC4035(<https://www.ietf.org/rfc/rfc4035.txt>). Besides I recommend you to read the NIST Secure Domain Name System (DNS) Deployment Guide, (NIST Special Publication 800-81-2) (<http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-81-2.pdf>). DNSSEC validates the source of the DNS response, ensures the response hasn't been altered in transit and authenticates replies of non-existence. It works by adding digital signatures to DNS responses, adding chain of trusts to validate responses and identifying bogus responses.

To facilitate signature validation, DNSSEC adds a few new DNS record types:

RRSIG – Contains a cryptographic signature

DNSKEY – Contains a public signing key



Participants



State Informative ()

Reported to [Skyliner](#)

Disclosed September 30, 2016 10:39pm +0400

Severity No Rating (---)

Weakness None

CVE ID None

Account de... None

Missing HTTP headers

6

#343928

Session Cookie Without Secure Flag

Share:

SUMMARY BY CYBERTIGER



Hello Everyone,
It's not a report.

In a comment, Ed @edoverflow said that the cookie Missing 'Secure' Flag requires XSS.
But I think that the cookie missing Secure Flag doesn't require XSS. It requires MITM(Man-In-The-Middle).
Cookie missing 'HttpOnly'Flag requires XSS.
If u think that I'm correct so plz vote up this report.
Thanks.

TIMELINE



cybertiger submitted a report to Ed.

Hi Ed,

The bug mentioned in the report [#343095](#) is not yet correctly patched I believe.

Apr 27th (3 years ago)

⋮

»

Reported April 27, 2018 4:50pm +0400

cybertiger

Participants



State Not-applicable ()

Reported to Ed

Disclosed April 28, 2018 3:44pm

Severity None (0.0)

Weakness None

CVE ID None

Account de... None

@davwwwx

16

#225833

www.hackerone.com website CSP "script-src" includes "unsafe-inline"

Share: [f](#) [t](#) [in](#) [Y](#) [e](#)

SUMMARY BY ROOTKID



The www.hackerone.com website was missing a strict "script-src" value. Albeit just a low profile risk, a sound "script-src" value should be considered best practice.

TIMELINE



rootkid submitted a report to [HackerOne](#).

May 3rd (4 years ago)

Summary:

The HTTP header of the hackerone.com website includes an unsafe CSP parameter for "script-src".

Description:

The hackerone.com website (<https://www.hackerone.com>) has a Content-Security-Policy configured, as pointed out on the Bug Bounty page of their program:

We utilize a strict Content Security Policy and a safe-by-default templating language to effectively neutralize Cross-Site Scripting (XSS).

Reported May 3, 2017 5:58pm +0400

rootkid

Participants



State Resolved ()

Reported to [HackerOne](#) Managed

Disclosed May 23, 2017 10:24am +0400

Severity None (0.0)

Weakness [None](#)

CVE ID [None](#)

Account de... [None](#)

3

#187225

Web Browser XSS Protection Not Enabled

Share:

TIMELINE



snicker2812 submitted a report to [Open-Xchange](#).

Dec 1st (5 years ago)

Web Browser XSS Protection is not enabled, or is disabled by the configuration of the 'X-XSS-Protection' HTTP response header on the web server

[http://www.dovecot.fi/s=..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5CWindows%5Csyste](http://www.dovecot.fi/s=..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5C..%5CWindows%5Csyste)
m.ini&submit=Search

<http://www.dovecot.fi/60/indexd7de.html?full-site=c%3A%2FWindows%2Fsystem.ini>

<http://www.dovecot.fi/?s=%2F..%2FWEB-INF%2Fweb.xml&submit=Search>

The X-XSS-Protection HTTP response header allows the web server to enable or disable the web browser's XSS protection mechanism. The following values would attempt to enable it:

X-XSS-Protection: 1; mode=block

X-XSS-Protection: 1; report=<http://www.example.com/xss>

The following values would disable it:

X-XSS-Protection: 0

The X-XSS-Protection HTTP response header is currently supported on Internet Explorer, Chrome and Safari (WebKit).

Reported December 1, 2016 10:27am +0400

[snicker2812](#)

Participants



State Not-applicable ()

Reported to [Open-Xchange](#)

Disclosed February 9, 2017 9:09am +0400

Severity Medium (4 ~ 6.9)

Weakness None

CVE ID None

Account de... None

Clickjacking

2

#1301113

CLICKJACKING LEADS TO DEACTIVATE ACCOUNT

Share:

TIMELINE



scianto05 submitted a report to [UPchieve](#).
Hello UPCHEIVE SECURITY TEAM,

Aug 12th (22 days ago)

I'm Anto

Vulnerability :

Clickjacking in (<https://hackers.upchieve.org/profile>)

Steps to Reproduce:

1). Create a HTML file with following code

```
<!DOCTYPE HTML>
```

Code 550 Bytes

```
1 <html lang="en-US">
2   <head>
3     <meta charset="UTF-8">
4   </head>
```

Wrap lines Copy Download

Reported August 12, 2021 11:03am +0400

scianto05

Participants



State

Duplicate ()

Reported to [UPchieve](#)

Disclosed August 16, 2021 9:21pm +0400

Severity

Low (0.1 ~ 3.9)

Weakness None

CVE ID None

Account de... None

127

#85624

Highly wormable clickjacking in player card

Share: [f](#) [t](#) [in](#) [Y](#) [e](#)

SUMMARY BY FILEDESCRIPTOR



<https://blog.innerht.ml/google-yolo/>

TIMELINE



filedescriptor submitted a report to [Twitter](#).

Aug 30th (6 years ago)

Hi,

I would like to report an issue where player card is vulnerable to clickjacking in certain browsers. This may result in something similar to XSS worm and many other critical damages.

Details

Twitter Player Card allows a website to embed a custom player(html) into an iframe in a tweet. There are currently 2-3 security features in place to defend clickjacking on Twitter:

1. `X-Frame-Options: SAMEORIGIN` covering the whole twitter.com domain
2. `Content-Security-Policy: frame-ancestors 'self'` ditto
3. JS-based frame-buster in some pages (but not all)

Reported August 30, 2015 10:43am +0400

filedescriptor

Participants



State Resolved ()

Reported to [Twitter](#)

Disclosed May 18, 2018 3:11am +0400

Severity No Rating (---)

Weakness None

Bounty \$5,040

CVE ID None

Account de... None

<https://hackerone.com/reports/85624>

@davwwwx

Weak cipher suites

3

#223350 Web server is vulnerable to Beast Attack

Share:

TIMELINE

mrr3boot submitted a report to [Weblate](#).

Apr 24th (4 years ago)

Supported versions:

TLSv1.0 TLSv1.1 TLSv1.2

Deflate compression: no

Supported cipher suites (ORDER IS NOT SIGNIFICANT):

TLSv1.0

RSA_WITH_AES_128_CBC_SHA

DHE_RSA_WITH_AES_128_CBC_SHA

Reported April 24, 2017 1:56pm +0400

mrr3boot

Participants



State Resolved ()

Reported to [Weblate](#)

Disclosed April 25, 2017 12:37am +0400

Severity Low (0.1 ~ 3.9)

Weakness None

CVE ID None

Account de... None

Host header injection

0

#170333

Host Header Injection/Redirection

Share: [f](#) [t](#) [in](#) [y](#) [p](#)

TIMELINE

#!

gorkhali submitted a report to [RubyGems](#).

Sep 19th (5 years ago)

rubygems.org is vulnerable to host header injection because the host header can be changed to something outside the target domain.

Attack vectors are somewhat limited but depends on how the host header is used by the back-end application code. If code references the hostname used in the URL such as password reset pages, an attacker could spoof the host header of the request in order to trick the application to forwarding the password reset email to the attackers domain instead, etc. Other attack vectors may also be possible through manipulation of hyperlinks or other misc. code that relies on the host/domain of the request.

nc rubygems.org 80

GET / HTTP/1.1

Host: google.com

HTTP/1.1 301 Moved Permanently

Server: nginx

Date: Mon, 19 Sep 2016 06:44:25 GMT

Content-Type: text/html

Transfer-Encoding: chunked

>>

Reported September 19, 2016 10:45am +0400

#! gorkhali

Participants

#! 

State  Duplicate ()

Reported to [RubyGems](#)

Disclosed February 9, 2018 3:15am +0400

Severity  None (0.0)

Weakness *None*

CVE ID *None*

Account de... *None*

67

#698416

Host Header Injection

Share:

TIMELINE

masterhacker submitted a report to New Relic.

Sep 20th (2 years ago)

Reproduction

1- open reset link <https://login.newrelic.com/passwords/forgot>

2- Enter the victim's email address and click Reset and Email Password

3- Intercept the HTTP request in Burp Suite & add X-Forwarded Host Header and write
attacker.com/.newrelic.com

link will be like

[https://testing-](https://testing-now.000webhostapp.com/.newrelic.com/passwords/reset/a248d8b06e7b25a116859729cbc0e07e180d9fb197dadc04f30185512eecc811)

[now.000webhostapp.com/.newrelic.com/passwords/reset/a248d8b06e7b25a116859729cbc0e07e180d9fb197dadc04f30185512eecc811](https://testing-now.000webhostapp.com/.newrelic.com/passwords/reset/a248d8b06e7b25a116859729cbc0e07e180d9fb197dadc04f30185512eecc811)

Impact

The victim will receive the malicious link in their email, and, when clicked, will leak the user's password reset link / token to the attacker, leading to full account takeover.

Reported September 20, 2019 4:49am +0400

masterhacker

Participants



State Resolved ()

Reported to New Relic

Disclosed January 27, 2020 8:13pm +0400

Severity Low (0.1 ~ 3.9)

Weakness None

Bounty \$500

CVE ID None

Account de... None

<https://hackerone.com/reports/698416>

@davwwwx

Information disclosure

Product, version

1

#23447

Version Disclosure (NginX)

Share:

TIMELINE



stalker submitted a report to Mail.ru.

Aug 10th (7 years ago)



POC:

url : <https://calendar.mail.ru>

Open up your google chrome browser.

Click right mouse button and choose Inspect Element.

Put website url in address bar. (<https://calendar.mail.ru>)

Now choose network option from Inspect Element menu.

Response Headers

Connection:close

Content-Security-Policy:default-src .mail.ru .imgsmail.ru .yadro.ru .facebook.com .vk.com .odnoklassniki.ru .tns-counter.ru .youtube.com;
 script-src 'unsafe-inline' 'unsafe-eval' .mail.ru .imgsmail.ru .yadro.ru .facebook.com .vk.com .odnoklassniki.ru .tns-counter.ru .youtube.com
 .twitter.com; style-src 'unsafe-inline' 'unsafe-eval' .mail.ru .imgsmail.ru .youtube.com; img-src data: *; report-uri

<https://cspreport.mail.ru/calendar/>;

Content-Type:text/html; charset=utf-8

Reported August 10, 2014 4:09pm +0400

stalker

Participants



State Informative ()

Reported to [Mail.ru](#)

Disclosed September 10, 2014 1:13pm +0400

Severity No Rating (---)

Weakness None

CVE ID None

Account de... None

5

#1245055

XSS DUE TO CVE-2020-3580

Share:

TIMELINE



veshrajghimire submitted a report to U.S. Dept Of Defense.

Hello Team,

During my research, I found the following host to be vulnerable to CVE 2020-3580 which is POST BASED XSS.

Jun 26th (2 months ago)



Vulnerable URL: [https://\[REDACTED\]/+CSCOE+/saml/sp/acs?tgname=a](https://[REDACTED]/+CSCOE+/saml/sp/acs?tgname=a)

Impact

Attackers can steal cookies and even takeover accounts and perform different malicious activities.

System Host(s)



Affected Product(s) and Version(s)

CVE Numbers

Steps to Reproduce

Save following code as xss.html and open in browser:

<https://hackerone.com/reports/1245055>



Reported June 26, 2021 3:42pm +0400

veshrajghimire

Participants



State Resolved ()

Reported to [U.S. Dept Of Defense](#)

Disclosed July 29, 2021 11:46pm +0400

Severity Medium (4 ~ 6.9)

Weakness None

CVE ID None

Account de... None

@davwwwx

21

#1003980

CVE-2020-14179 on https://jira.theendlessweb.com/secure/QueryComponent!Default.jspa leads to information disclosure

Share:

SUMMARY BY NAGLI



Please do not report this issue to paying programs, it's up to their decision to fix it as it won't disclose any sensitive information.

You will most likely get N/A or Informative from this CVE.

Take care!

TIMELINE



nagli submitted a report to [Endless Group](#).

Hello theendlessweb team,

Oct 10th (11 months ago)

Summary:

the Jira instance on jira.theendlessweb.com is vulnerable to [CVE-2020-14179](#) which allows remote, unauthenticated attackers to view custom field names and custom SLA names via an Information Disclosure vulnerability

<https://hackerone.com/reports/1003980>

Reported October 10, 2020 12:40am +0400

nagli

Participants



State Resolved ()

Reported to [Endless Group](#)

Disclosed November 20, 2020 11:23am +0400

Severity Medium (4 ~ 6.9)

Weakness [None](#)

CVE ID [None](#)

Account de... [None](#)

@davwwwx

Directory listing

8

#690796

Directory listing is enabled that exposes non public data through multiple path

Share:

TIMELINE



tibin_sunny submitted a report to [Nextcloud](#).

Sep 9th (2 years ago)

Directory Listing is enabled on <https://try.nextcloud.com> and it shows out a few files on the server + The server version.

POC: <https://try.nextcloud.com/assets/>

<https://try.nextcloud.com/css/>

<https://try.nextcloud.com/js/>

Impact

This could leak sensitive information on the server and it also allows an attacker to gain knowledge about the web-technology used by the website

1 attachment:

F578277: [nextcloud.PNG](#)

Reported September 9, 2019 12:59pm +0400

[tibin_sunny](#)

Participants



State Resolved ()

Reported to [Nextcloud](#)

Disclosed February 1, 2020 8:39am +0400

Severity Low (0.1 ~ 3.9)

Weakness [None](#)

CVE ID [None](#)

Account de... [None](#)

Verbose errors

1

#1082521

Full Path Disclosure of Server through 500 Server Error

Share: [f](#) [t](#) [in](#) [Y](#) [m](#)

TIMELINE



bugera submitted a report to Kartpay.

Hello team,

Jan 20th (8 months ago)

EXPLANATION

I found a interesting vulnerability into your site that it unexpected disclosing the server path where the PHP files are being hosted. When application sends account verification links in email then if anyone tries to verify his account with that link at a twice then on the title of the website the whole server path is disclosing through 500 Server Error.

Vulnerable Path :

```
/usr/share/nginx/website/resources/view/auth/create_password.blade.php
```

I have added a POC .

Impact

1. Server Information Disclosure

Reported January 20, 2021 7:41pm +0400

[bugera](#)

Participants



State [Resolved \(\)](#)

Reported to [Kartpay](#)

Disclosed August 16, 2021 9:46pm +0400

Severity Low (0.1 ~ 3.9)

Weakness [None](#)

CVE ID [None](#)

Account de... [None](#)

193

#1083543

Debug Mode Leak Critical Information [AWS Keys , SMTP , Database , Django Secret Key (RCE) , Dodoc , Telegram , Twilio ..]

Share: [f](#) [t](#) [in](#) [y](#) [g](#)

SUMMARY BY MAIL.RU



Debug mode was enabled in legium-back.corp.mail.ru leaking some potentially sensitive information

SUMMARY BY YUKUSAWA18



Debug mode was enabled in legium-back.corp.mail.ru leaking some potentially sensitive information

TIMELINE



yukusawa18 submitted a report to [Mail.ru](#).

Jan 22nd (7 months ago)



kpebetka Mail.ru staff closed the report and changed the status to ● Not Applicable.

Jan 22nd (7 months ago)



yukusawa18 posted a comment.

Updated Jan 22nd (7 months ago)

»
Reported January 22, 2021 12:18am +0400

yukusawa18

Participants



State ● Resolved ()

Reported to [Mail.ru](#)

Disclosed May 24, 2021 1:29pm +0400

Severity Critical (9 ~ 10)

Weakness *None*

Bounty \$7,500

CVE ID *None*

Account de... *None*

<https://hackerone.com/reports/1083543>

@davwwwx

Leaks in services

21

#631529

Listing of Amazon S3 Bucket accessible to any amazon authenticated user (vector-maps-e457472599)

Share: [f](#) [t](#) [in](#) [y](#) [d](#)

TIMELINE



zerOttl submitted a report to TomTom.

Jun 28th (2 years ago)

Reported June 28, 2019 6:48pm +0400

zerOttl

Participants



State ● Resolved ()

Reported to [TomTom](#) Managed

Disclosed August 9, 2019 4:38pm +0400

Severity ■■■ Medium (4 ~ 6.9)

Weakness *None*

CVE ID *None*

Account de... *None*

Summary:

It's possible to get a listing of every files in the S3 bucket [vector-maps-e457472599](#)

Description:

The problem is using the AWS command line, it's possible to get a listing of files in the Amazon S3 Bucket with an AWS authentication. See screenshot [vector-maps-e457472599_public_s3_bucket.png](#)

This user authentication is easy to get and it's free from Amazon.

The good news is that the ACL on the files are set the way that's impossible at moment to create any file from the bucket using my authentication. I did not test removing any file from the bucket.

A secure amazon S3 bucket would show Access Denied like your other bucket named [brda-vector-maps](#) in screenshot [brda-vector-maps_access_denied_s3_bucket.png](#)

92

#911606

Leaked JFrog Artifactory username and password exposed on GitHub - <https://snapchat.jfrog.io>

Share:

SUMMARY BY SNAPCHAT



Researcher found valid jFrog credentials which were committed to a public Github repository of a Snap employee. This allowed access to internal Snap libraries/artifacts along with the ability to push updates to existing artifacts as well.

TIMELINE



kiyell submitted a report to [Snapchat](#).

Jun 30th (about 1 year ago)



bugtriage-ryan posted a comment.

Jun 30th (about 1 year ago)



sfrisk Snapchat staff changed the status to [Triaged](#).

Jun 30th (about 1 year ago)

»

Reported June 30, 2020 9:00am +0400

[kiyell](#)

Participants



State [Resolved \(\)](#)

Reported to [Snapchat](#)

Disclosed August 13, 2021 1:40am +0400

Severity High (7 ~ 8.9)

Weakness [None](#)

Bounty \$15,000

<https://hackerone.com/reports/911606>

@davwwwx

Content spoofing on error pages

»	
Reported November 11, 2016 8:51pm +0400	
 ak1t4	
Participants	
 	
State	● Resolved ()
Reported to	LocalTapiola
<hr/>	
Disclosed	January 9, 2017 1:12pm +0400
Severity	 Low (0.1 ~ 3.9)
Weakness	<i>None</i>
Bounty	\$100
<hr/>	
CVE ID	<i>None</i>
Account de...	<i>None</i>

Wordpress

XMLRPC enabled

29

#448524

xmlrpc.php file is enable it will used for (DOS) and bruteforce attack

Share:

TIMELINE



meepmerp submitted a report to [FormAssembly](#).

Nov 21st (3 years ago)

Wordpress that have xmlrpc.php enabled for pingbacks, trackbacks, etc. can be made as a part of a huge botnet causing a major DDOS. The website <https://www.formassembly.com/> has the xmlrpc.php file enabled and could thus be potentially used for such an attack against other victim hosts.

In order to determine whether the xmlrpc.php file is enabled or not, using the Repeater tab in Burp, send the request below.

POST /wp/xmlrpc.php HTTP/1.1

Host: www.formassembly.com

User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:52.0) Gecko/20100101 Firefox/52.0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,/;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Connection: close

Upgrade-Insecure-Requests: 1

Cache-Control: max-age=0

Reported November 21, 2018 9:29pm +0400

meepmerp

Participants



State Resolved ()

Reported to [FormAssembly](#)

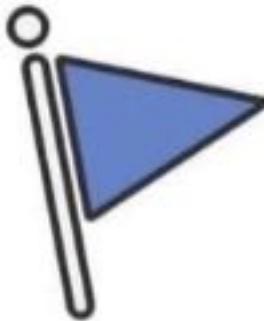
Disclosed December 27, 2018 7:55am +0400

Severity High (7 ~ 8.9)

Weakness None

CVE ID None

Account de... None



Marked Safe From
XMLRPC.php open
Today

wp-json origin reflection

141

#768151

Bypassing CORS Misconfiguration Leads to Sensitive Exposure

Share:

TIMELINE



koaladev submitted a report to [U.S. Dept Of Defense](#).

Jan 4th (2 years ago)

Hi! Security Team [@deptofdefense](#),

It's possible to get information about the users registered (such as: id, name, login name, etc.) without authentication in Wordpress via API on
*. [REDACTED].

Description:

By default Wordpress allow public access to Rest API to get informations about all users registered on the system.

Platform(s) Affected: [website]

*.https://[REDACTED]/wp-json/

Steps To Reproduce:

- 1) Repreat URL Vulnerable to Burp Suite
- 2) If you add the `Origin-parameter` to the `Request-header`, the responsive header will reject
- 3) Bypassing Using Exploit CORS-With Sensitive

Reported January 4, 2020 7:52pm +0400

koaladev

Participants



State Resolved ()

Reported to [U.S. Dept Of Defense](#)

Disclosed May 14, 2020 9:16pm +0400

Severity Medium (4 ~ 6.9)

Weakness *None*

CVE ID *None*

Account de... *None*

Open redirect

50

#1073565

Open Redirect on https://www.twitterflightschool.com/widgets/experience?destination_url=https://evil.com

Share: [f](#) [t](#) [in](#) [y](#) [p](#)

SUMMARY BY TWITTER



This report details an open redirect issue that enabled crafting potentially malicious URLs which could be used to redirect users to a site specified in a URL parameter of the URL creator's choosing. This may allow an attacker to exploit a user's trust by leveraging open redirect on the affected subdomains (flightschool.twitter.com and takeflight.twitter.com) to launch phishing scams by masquerading as the legitimate websites.

SUMMARY BY NAGLI



This vulnerability is part of a study I have been conducting about "Vulnerability inheritance - Attacking companies and scoring bounties through 3rd party integrations".

twitterflightschool.com was pointing towards a 3rd party vendor as a CNAME, I managed to find Stored Open Redirect on an endpoint which belongs to the vendor that allowed me to achieve Stored Open Redirects on 2 domains under *.twitter.com which are

Code 26 Bytes

```
1 flightschool.twitter.com,
```

»

Reported January 7, 2021 8:14pm +0400

nagli

Participants



State Resolved ()

Reported to Twitter

Disclosed May 5, 2021 1:09am +0400

Severity Low (0.1 ~ 3.9)

Weakness None

CVE ID None

Account de... None

Open redirectors

Open redirectors take you from a Google URL to another website chosen by whoever constructed the link. Some members of the security community argue that the redirectors aid phishing, because users may be inclined to trust the mouse hover tooltip on a link and then fail to examine the address bar once the navigation takes place.

Our take on this is that tooltips are not a reliable security indicator, and can be tampered with in many ways; so, we invest in technologies to detect and alert users about phishing and abuse, but we generally hold that a small number of properly monitored redirectors offers fairly clear benefits and poses very little practical risk.

Of course, some improperly designed redirectors can lead to more serious flaws, and we often see it used to trigger the following vulnerabilities:

- Content Security Policy bypass
- Referrer check bypass
- URL whitelist bypass
- Angular ng-include bypass
- Working redirect to *javascript: URL*

If you notice the above issues, use the found open redirector in the exploit chain and let us know! On its own though, the open redirector will not be accepted for the VRP.

<https://sites.google.com/site/bughunteruniversity/nonvuln/open-redirect>



123

#206591

Open Redirect on central.uber.com allows for account takeover

Share: [f](#) [t](#) [in](#) [y](#) [d](#)

SUMMARY BY UBER



An error in our OAuth2 flow for `central.uber.com` allowed an attacker to leverage an open redirect that allowed for a full account takeover. When logging into `central.uber.com`, the `state` parameter for `login.uber.com` contained a redirect location instead of a CSRF token. As a result, an attacker could modify the `state` parameter to have a poisoned `central.uber.com` path which would redirect to a custom domain after login and allow them to steal an account OAuth access token.

Thanks, [@ngalog!](#)

TIMELINE

 ngalog submitted a report to Uber .	Feb 15th (5 years ago)
 ngalog updated the severity to Critical.	Feb 15th (5 years ago)
 ngalog posted a comment.	Updated Feb 15th (5 years ago)
 jovon-uber posted a comment.	Feb 15th (5 years ago)

Reported February 15, 2017 1:48pm +0400



ngalog

Participants



VATAP

State ● Resolved ()Reported to [Uber](#) Managed

Disclosed January 25, 2019 9:41pm +0400

Severity High (7 ~ 8.9)Weakness None

Bounty \$8,000

CVE ID NoneAccount de... None

Custom data

@davwwwx

Server Side Request Forgery

18

#145524

Server side request forgery (SSRF) on nextcloud implementation.

Share: [f](#) [t](#) [in](#) [y](#)

TIMELINE



paglababa submitted a report to [Nextcloud](#).

Jun 17th (5 years ago)

An admin of nextcloud server can add other trusted nextcloud server in his own installation. The following request passes when a new add^{*} request is processed:

Code 829 Bytes

[Wrap lines](#) [Copy](#) [Download](#)

```

1 POST /nextcloud/index.php/apps/federation/trusted-servers HTTP/1.1
2 Host: myown.nextcloudserver.com
3 User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.11; rv:47.0) Gecko/20100101 Firefox/47.0
4 Accept: /*
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Content-Type: application/x-www-form-urlencoded; charset=UTF-8
8 requesttoken: GAFcYDUGGyM0CCYeIlk4b19ADhw0FgcL0y4kERdDL1Q=:AL1VmGJMGqQsVhw59y9yE/wsJGJWMtc8DJljuFMaI4=
9 OCS-APIREQUEST: true
10 X-Requested-With: XMLHttpRequest
11 Content-Length: 27
12 Cookie: oc6wp9sjado5=nnofa4hfq2esn7anu70hg3c2h0; oc_sessionPassphrase=dvniWxtCrcQk4Nbt4eXXmyZu5wUk3JoHziCUaCBcmeQFaM0333bS8HBwvFO
13 Connection: close

```

»

Reported June 17, 2016 11:27pm +0400

[paglababa](#)

Participants



State Informative ()

Reported to [Nextcloud](#)

Disclosed June 17, 2016 11:41pm +0400

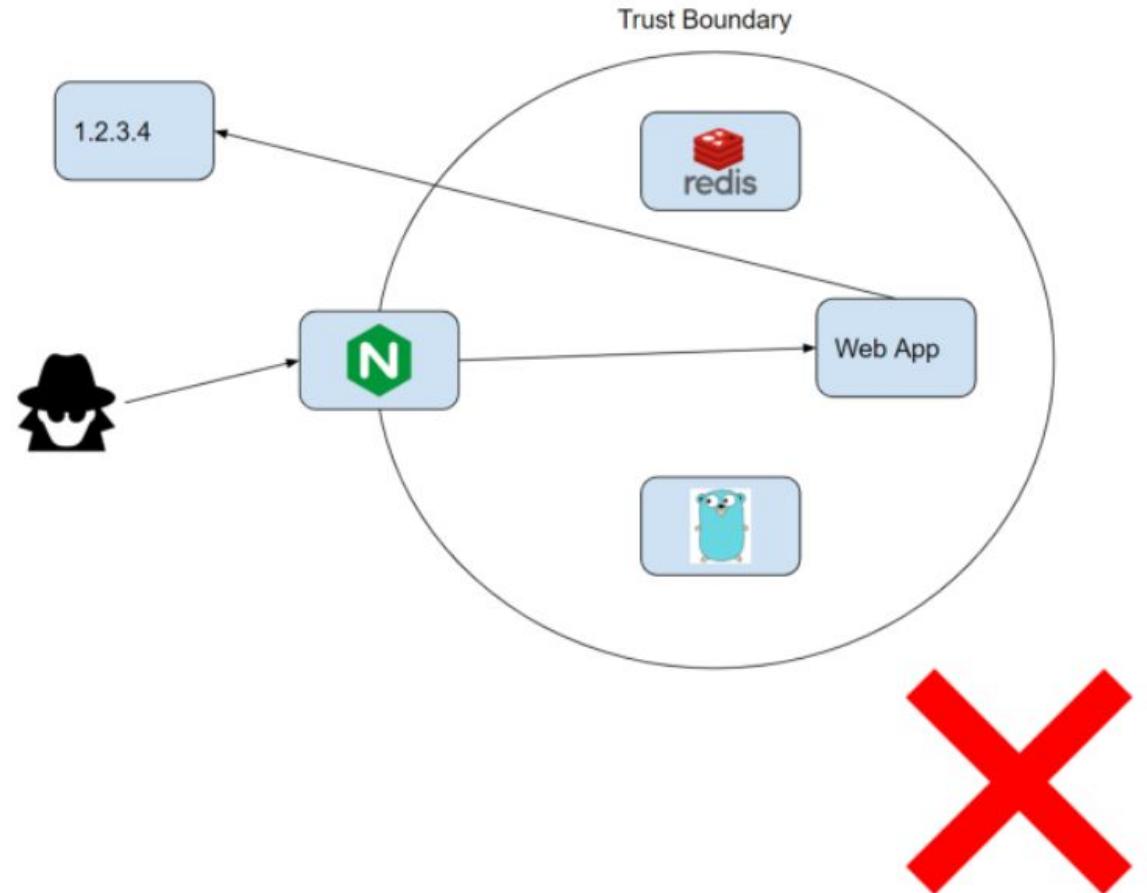
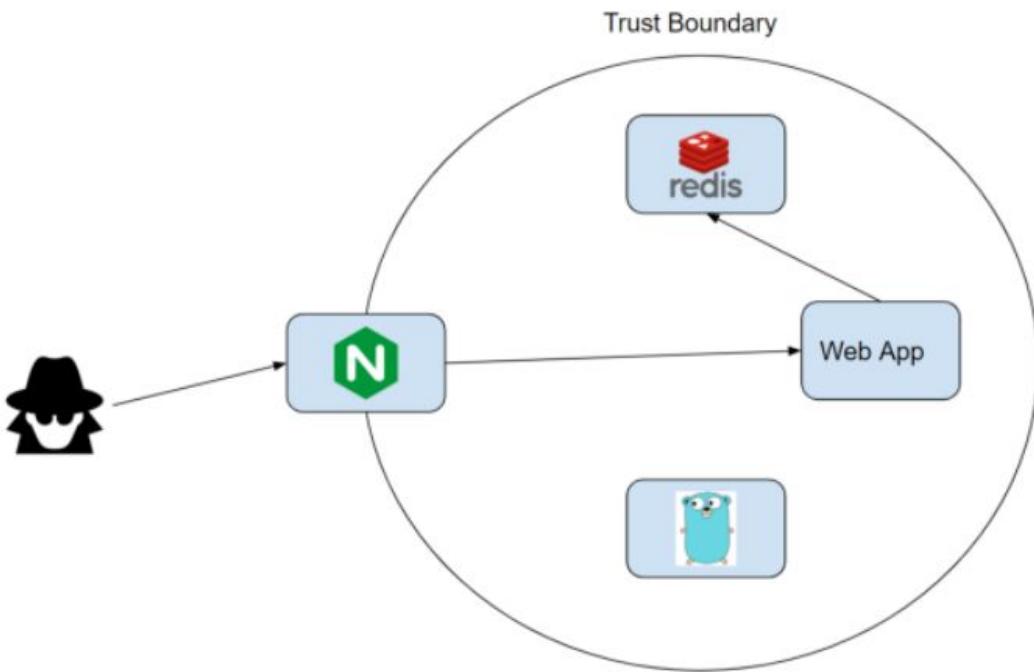
Severity No Rating (---)

Weakness [None](#)

CVE ID [None](#)

Account de... [None](#)

@davwwwx



Impactful SSRF on left; not right

<https://medium.com/@d0nut/piercing-the-veal-short-stories-to-read-with-friends-4aa86d606fc5>

@davwwwx

340

#530974

Server-Side Request Forgery using Javascript allows to exfill data from Google Meta data

Share:

SUMMARY BY SNAPCHAT



@nahamsec, @daeken and @ziot found a Server-Side Request Forgery (SSRF) vulnerability in <https://business.snapchat.com> which they exploit by providing a custom webpage configured to utilize DNS rebinding to access internal web endpoints like the Google Metadata Service. Using this they are able to mint tokens for the service-account assigned to the instance hosting the Chrome instances used for extracting webpages assets for media projects.

TIMELINE



nahamsec submitted a report to [Snapchat](#).

Apr 8th (2 years ago)

Hey there,

I was looking at your ads site with @daeken, we found some weird behavior in the import function of the creative app. Here are the steps:

POC

- Login to <https://business.snapchat.com/>
- Go to creative library -> New Creative
- Under "Topsnap Media", click on "Create"

<https://hackerone.com/reports/530974>

»

Reported April 8, 2019 9:29am +0400

nahamsec

Participants



State Resolved ()

Reported to [Snapchat](#)

Disclosed November 30, 2020 10:27pm +0400

Severity No Rating (---)

Weakness [None](#)

Bounty \$4,000

CVE ID [None](#)

Account de... [None](#)

@davwwwx

XSS

SELF XSS

19

#846931

XSS at go.mail.ru

Share: [f](#) [t](#) [in](#) [y](#) [g](#)

SUMMARY BY MAIL.RU



DOM-based self XSS in go.mail.ru social search functionality

TIMELINE



adiosmf submitted a report to [Mail.ru](#).

Apr 11th (about 1 year ago)



kpebetka (Mail.ru staff) changed the status to ● Triaged.

Apr 11th (about 1 year ago)



Mail.ru has decided that this report is not eligible for a bounty.

Apr 11th (about 1 year ago)



majes7ic (Mail.ru staff) closed the report and changed the status to ● Resolved.

Apr 22nd (about 1 year ago)



adiosmf posted a comment.

Apr 22nd (about 1 year ago)

»

Reported April 11, 2020 2:21am +0400



adiosmf

Participants



State

● Resolved ()

Reported to

[Mail.ru](#)

Disclosed

May 8, 2020 2:49pm +0400

Severity

Medium (6.1)

Weakness

None

CVE ID

None

Account de...

None

@davwwwx

57

#1028332

Stored XSS on https://events.hackerone.com

Share: [f](#) [t](#) [in](#) [y](#)

SUMMARY BY HACKERONE

h1

@nagli found a stored Cross-Site Scripting vulnerability in a 3rd party vendor that was used by HackerOne. This system did not contain any data related to reports submitted and stored on hackerone.com. HackerOne worked with the vendor to remediate the vulnerability. The report is partially disclosed to anonymize the vendor.

SUMMARY BY NAGLI



This vulnerability is part of a study I have been conducting about "Vulnerability inheritance - Attacking companies and scoring bounties through 3rd party integrations".

events.hackerone.com was pointing towards a 3rd party vendor as a CNAME, I managed to find authenticated stored XSS on an endpoint which belongs to the vendor which allowed me to achieve RXSS through CSRF+Self XSS on Hackerone's subdomain.

TIMELINE



nagli submitted a report to HackerOne.

Nov 6th (10 months ago)

<https://hackerone.com/reports/1028332>

Reported November 6, 2020 5:33pm +0400



nagli

Participants



State

Resolved ()

Reported to

HackerOne Managed

Disclosed

March 27, 2021 12:25am +0400

Severity

None (0.0)

Weakness

None

CVE ID

None

Account de...

None

@davwwwx

53

#472470

[manage.jumpbikes.com] Blind XSS on Jump admin panel via user name

Share:

SUMMARY BY UBER



By setting a user's name to an XSS payload, a user was able to inject JavaScript which was executed on the administrative panel for Jump bikes, allowing complete compromise of the panel, exposing user activity, personal information and billing information.

TIMELINE



cablej submitted a report to Uber.

Dec 27th (3 years ago)



lindsey-uber posted a comment.

Dec 27th (3 years ago)

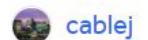


lindsey-uber changed the status to ● Triage.

Dec 29th (3 years ago)

»

Reported December 27, 2018 9:34am +0400



cablej

Participants



State

● Resolved ()

Reported to

Uber Managed

Disclosed

February 24, 2021 3:45am +0400

Severity

Critical (9 ~ 10)

Weakness

None

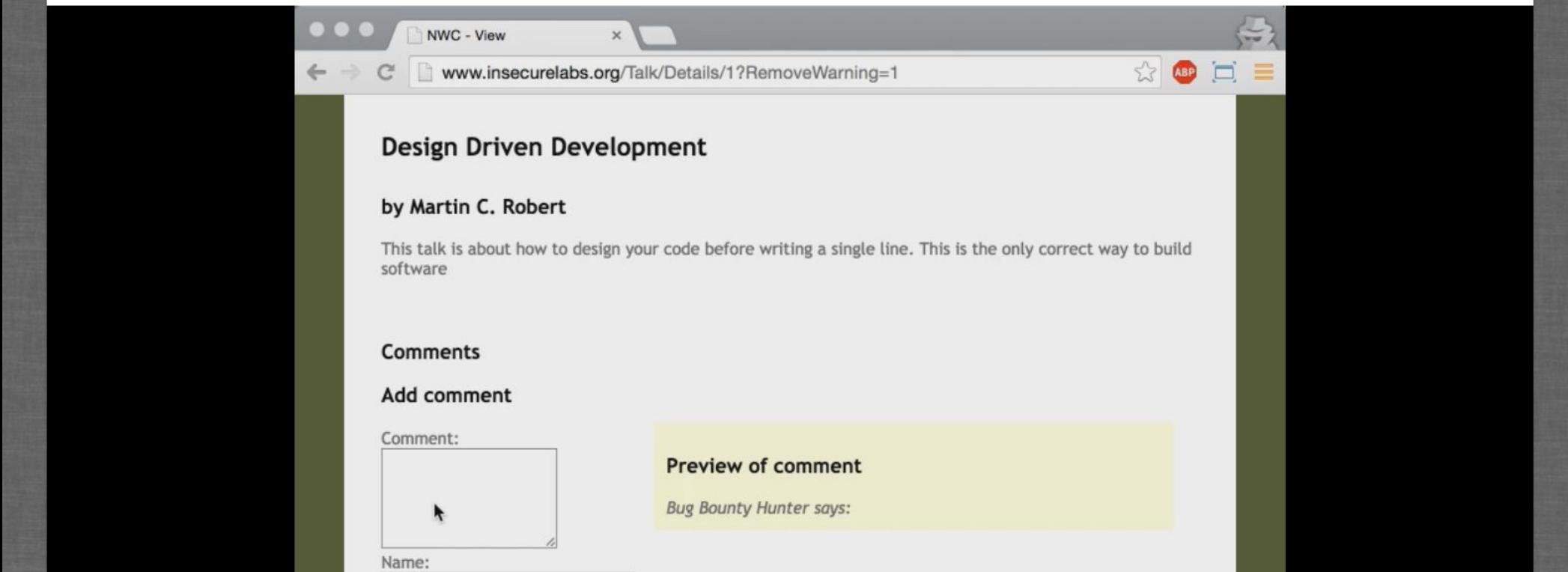
Bounty

\$4,000

<https://hackerone.com/reports/472470>

@davwwwx

If this is how you hunt for Cross-Site Scripting (XSS)...



<https://xsshunter.com/>

Cookie based XSS

2

#83576 [start.icq.com] Reflected XSS via Cookies

Share:

TIMELINE

bigbear_ submitted a report to [Mail.ru](#).

Aug 20th (6 years ago)

Request:

GET / HTTP/1.1

Cookie: geo=380; icqsrch_lang=ua; abt=1"><script>alert(document.domain) </script> <a href="; icq_pref=medium%3A_blank

Referer: <http://start.icq.com/>

Host: start.icq.com

Connection: Keep-alive

Accept-Encoding: gzip,deflate

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.21 (KHTML, like Gecko) Chrome/41.0.2228.0 Safari/537.21

Accept: /

Response:

[Code](#) 619 Bytes[Wrap lines](#) [Copy](#) [Download](#)

```
1 <div class="d3-1-3" id="icq_ads" onmouseover="showIcqAd('block')" onmouseout="showIcqAd('none')">
```

Reported August 20, 2015 7:29am +0400



bigbear_

Participants



State Resolved ()

Reported to [Mail.ru](#)

Disclosed October 21, 2015 3:27pm +0400

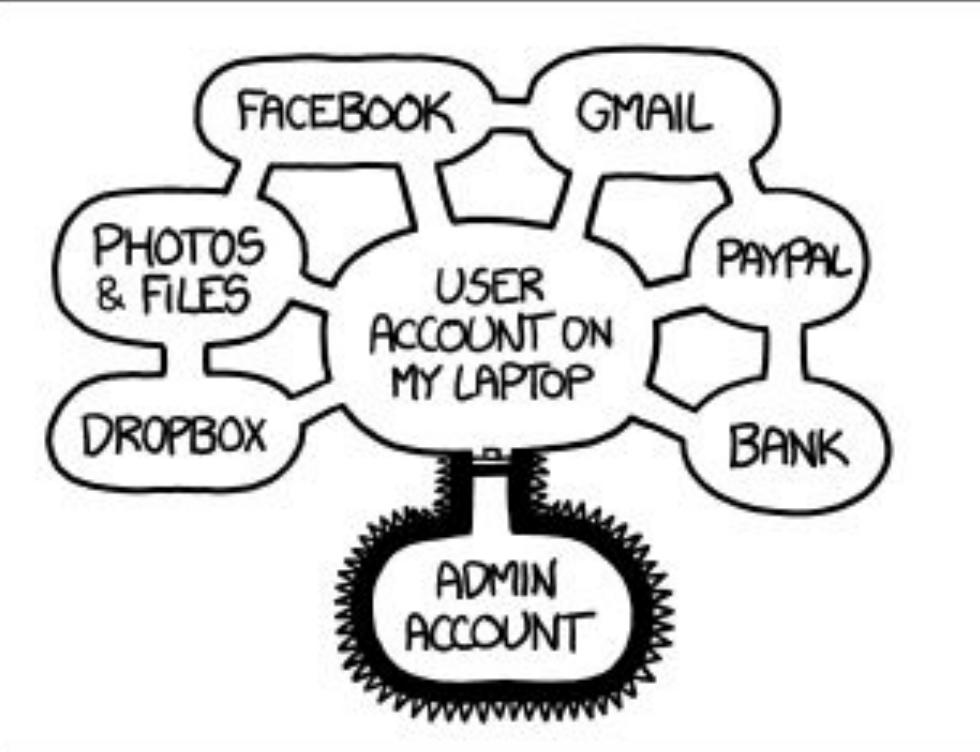
Severity No Rating (---)

Weakness None

CVE ID None

Account de... None

@davwwwx



IF SOMEONE STEALS MY LAPTOP WHILE I'M LOGGED IN, THEY CAN READ MY EMAIL, TAKE MY MONEY, AND IMPERSONATE ME TO MY FRIENDS,

BUT AT LEAST THEY CAN'T INSTALL DRIVERS WITHOUT MY PERMISSION.

<https://xkcd.com/1200/>

@davwwwx

Welcome unknownUser416

Every single day, we have a thirst to quench , a hunger to sate , an XSS to find . We all have our recipes , our payloads , that we cook up in order to find some bugs .

Today, Initigrity presents the **XSS cookbook** .

A way for you to organize and share your XSS payloads with the world! 
We've already added a couple in the list down below , but don't be afraid to experiment with our cool way of making recipe objects! 

The collection:

-
-
-

The basic XSS
The SVG
The POLYGLOT

Recipe: undefined

Ingredients:

Payload:

Steps:

<https://challenge-0821.intigrity.io/>

Sandbox domain XSS

XSS in sandbox domains

Google uses a range of [sandbox domains](#) to safely host various types of user-generated content. Many of these sandboxes are specifically meant to isolate user-uploaded HTML, JavaScript, or Flash applets and make sure that they can't access any user data.

For this reason, we recommend using `alert(document.domain)` instead of `alert(1)` as your default XSS payload. In particular, if you see script execution in any subdomains of the domains in this list:

- `ad.doubleclick.net`
- `googleusercontent.com`
- `googlecode.com`
- `codespot.com`
- `feeds.feedburner.com`
- `googleadservices.com`
- `googledrive.com`
- `googlegroups.com`
- `{your-blog-name}.blogspot.com`
- `{your-app-name}.appspot.com`
- `firebasestorage.googleapis.com`
- `storage.googleapis.com`
- `kaggleusercontent.com`
- `translate.goog`

...your report will probably not qualify, unless you can come up with an [attack scenario](#) where the injected code could gain access to sensitive user data.

<https://sites.google.com/site/bughunteruniversity/nonvuln/xss-in-sandbox-domain>

<https://sites.google.com/site/bughunteruniversity/nonvuln/>



BOUNTY PLZ !!!

Thanks !

<https://go.xss.am/bbs-owasp>