v1

Bitcoin Web Application Security

Bryan Bishop <kanzure@gmail.com>

Breaking Bitcoin Amsterdam 2019

0E4C A12B E16B E691 56F5 40C9 984F 10CC 7716 9FD2

whoami, background slide

- Bryan Bishop
- Software development background, lately: consulting + security audits
- Previously @ LedgerX (4 years)
- Bitcoin Core contributor
- Transcripts
- Biotech projects
- Follow me on twitter, or don't: https://twitter.com/kanzure

Web application security, in general

- Let's play a game
  - Defensive: protect site operations, user data, and user browsers from advanced persistent threats from all over the world.
  - Offensive: wait for someone to make a mistake.
- Asymmetry in defense vs offense
- Bitcoin brings a lot more attention and financial incentive to hack a site
  - Not your keys, not your coins
  - Don't trust, verify

Condensed cheat sheet for web hacking

https://github.com/OWASP/CheatSheetSeries/blob/master/Index.md

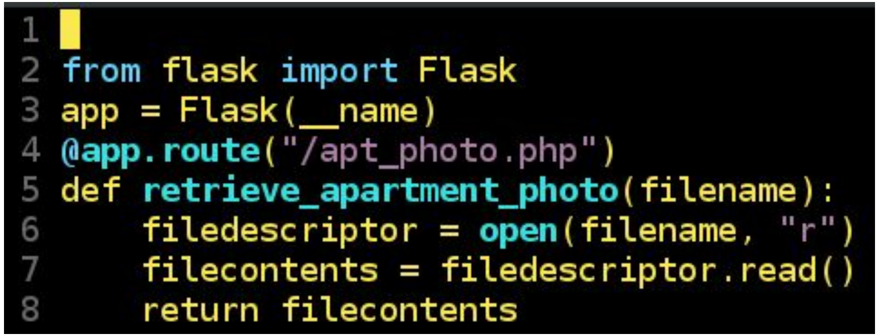
Injection, exfiltration, XML external entities, (de)serializer bugs (pickle/yaml), access control bugs, cross-site scripting (XSS), innerHtml (XSS), XML injection during construction, cross-site request forgery (CSRF), password hashing errors, password storage, file disclosure, brute force attacks, oauth flaws, clickjacking, content security policy (CSP) misconfiguration, credential stuffing (bruteforce), cookie jacking, poor key rotation schedules, denial of service vulnerabilities, docker daemon socket exposure, account recovery goofs, log exposure, ...

Example: File inclusion vulnerability

- You see a URL like:

  https://www.apartmentportal.com/apt_photo.php?name=IMG_50341.JPG

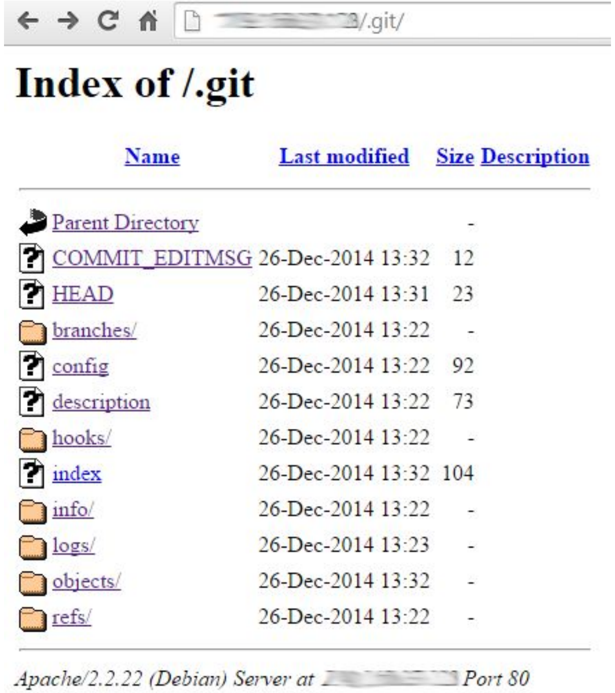
- Really, people used to make sites like this. It would look like this under the hood (python example):



```python

from flask import Flask
app = Flask(__name)
@app.route("/apt_photo.php")
def retrieve_apartment_photo(filename):
    filedescriptor = open(filename, "r")
    filecontents = filedescriptor.read()
    return filecontents
```

Example: Source code disclosure via exposed .git/

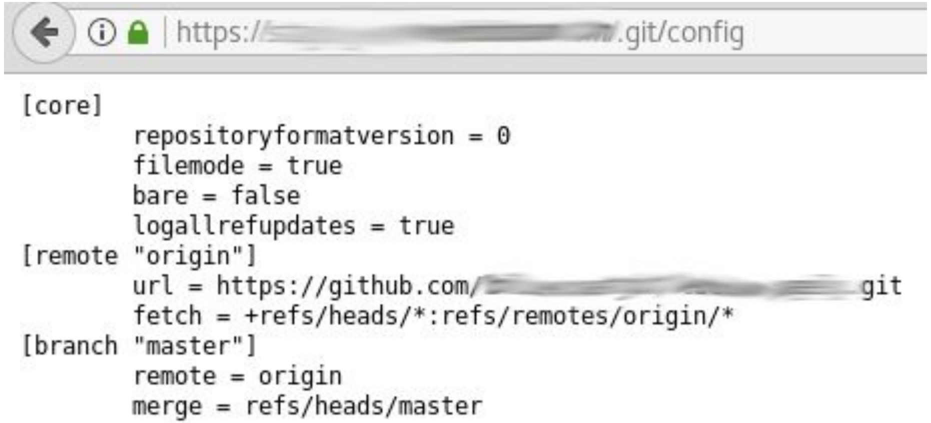


[redacted]/.git/

Index of /.git

Name	Last modified	Size	Description
Parent Directory		-	
COMMIT_EDITMSG	26-Dec-2014 13:32	12	
HEAD	26-Dec-2014 13:31	23	
branches/	26-Dec-2014 13:22	-	
config	26-Dec-2014 13:22	92	
description	26-Dec-2014 13:22	73	
hooks/	26-Dec-2014 13:22	-	
index	26-Dec-2014 13:32	104	
info/	26-Dec-2014 13:22	-	
logs/	26-Dec-2014 13:23	-	
objects/	26-Dec-2014 13:32	-	
refs/	26-Dec-2014 13:22	-	

Apache/2.2.22 (Debian) Server at [redacted] Port 80



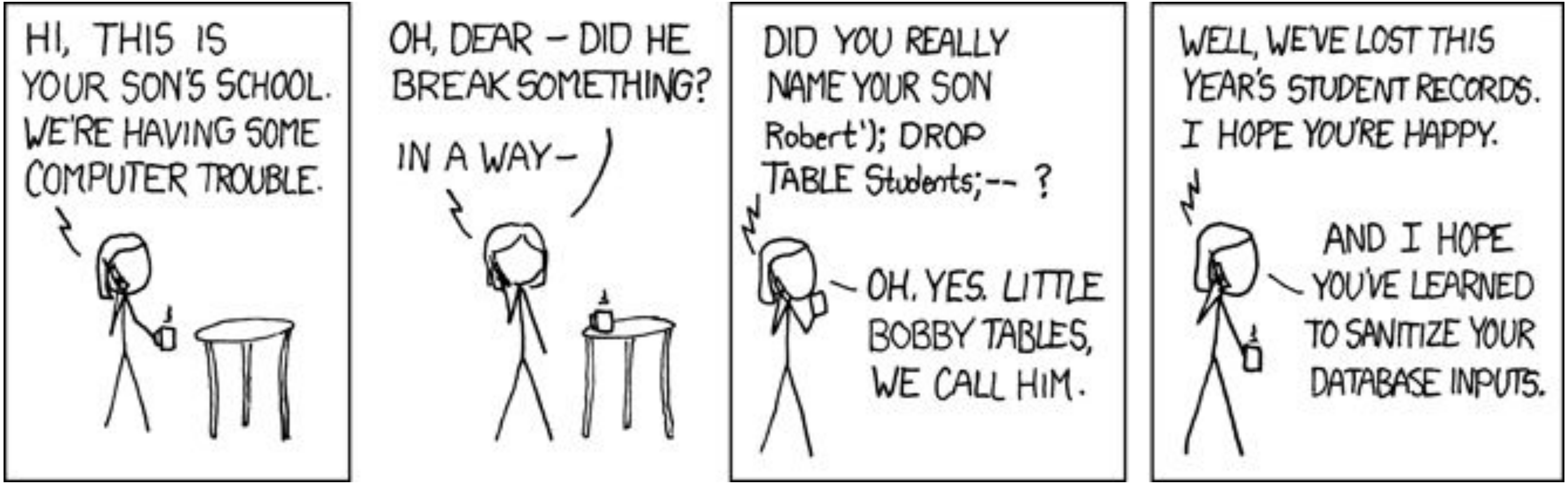
https:/[redacted].git/config

```
[core]
        repositoryformatversion = 0
        filemode = true
        bare = false
        logallrefupdates = true
[remote "origin"]
        url = https://github.com/[redacted]git
        fetch = +refs/heads/*:refs/remotes/origin/*
[branch "master"]
        remote = origin
        merge = refs/heads/master
```

Story time...

- The year was 2002. I was 12 years old. Little Bryan developed a somewhat popular 1000-user website about anime, a topic which I knew absolutely nothing about.
    - This website was a PHP monstrosity mixing together concepts from online RPGs, forums and other features I had no experience building.
- One day, incredibly bored at school, I wondered how to update my website from the school computers. No SSH. No FTP. Hmm.
    - After a lot of thinking, I came up with an idea where I could manipulate URL parameters to modify the database, and "only I would know about it".
- Yeah....... that's SQL injection.

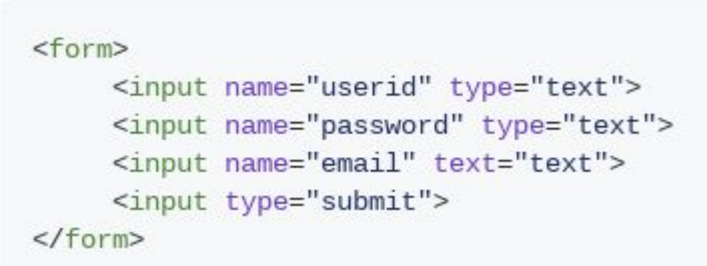
xkcd 327: Little Bobby Tables


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.

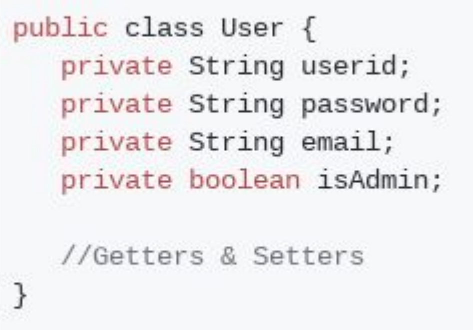
Example: Egor Homakov, GitHub and Rails incident

- Ruby on Rails
- Mass assignment vulnerability
- Targeted the Rails repository on GitHub
- Overwrote a developer's key, postdated an issue to 1001 years in the future, and pushed source code.

https://arstechnica.com/information-technology/2012/03/hacker-commandeers-github-to-prove-vuln-in-ruby/

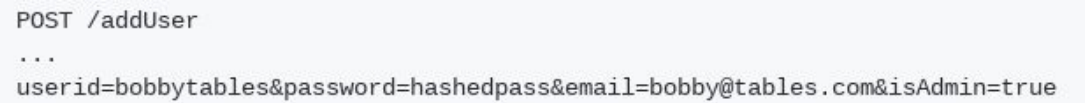


```html
<form>
     <input name="userid" type="text">
     <input name="password" type="text">
     <input name="email" text="text">
     <input type="submit">
</form>
```

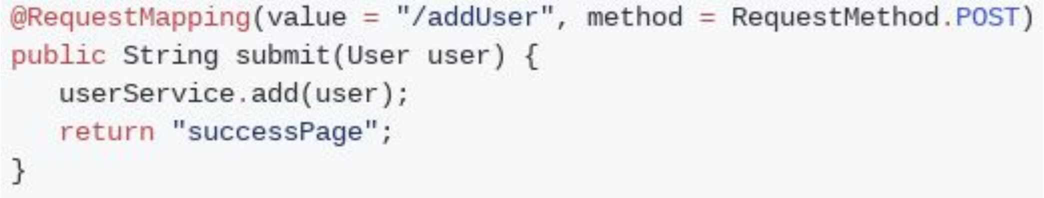


```java
public class User {
   private String userid;
   private String password;
   private String email;
   private boolean isAdmin;

   //Getters & Setters
}
```



```
POST /addUser
...
userid=bobbytables&password=hashedpass&email=bobby@tables.com&isAdmin=true
```



```java
@RequestMapping(value = "/addUser", method = RequestMethod.POST)
public String submit(User user) {
   userService.add(user);
   return "successPage";
}
```

Example: Cross-site request forgery (CSRF)

- Simplewallet (Monero wallet) hosted an RPC service on localhost port 18082
- API actions require no authorization, had no CSRF mitigation
- Websites simply had to use a localhost XMLHttpRequest (XHR) to access the wallet and spend coins

https://labs.mwrinfosecurity.com/advisories/csrf-vulnerability-allows-for-remote-compromise-of-monero-wallets/

Bitcoin exchanges

- Quote from a colleague about MtGox- "Yeah, I was poking around on their site and found a lot of holes. I really just waltzed in. This must have been 2010 or 2011." While he was probably making this up, it's entirely plausible.

Bitcoin exchanges

- Quote from a colleague about MtGox- "Yeah, I was poking around on their site and found a lot of holes. I really just waltzed in. This must have been 2010 or 2011." While he was probably making this up, it's entirely plausible. **HACKED**

Bitcoin exchanges

- Quote from a colleague about MtGox- "Yeah, I was poking around on their site and found a lot of holes. I really just waltzed in. This must have been 2010 or 2011." While he was probably making this up, it's entirely plausible. **HACKED**
- Alleged infiltration of Cryptsy

Bitcoin exchanges

- Quote from a colleague about MtGox- "Yeah, I was poking around on their site and found a lot of holes. I really just waltzed in. This must have been 2010 or 2011." While he was probably making this up, it's entirely plausible. **HACKED**
- Alleged infiltration of Cryptsy **HACKED**

Bitcoin exchanges

- Quote from a colleague about MtGox- "Yeah, I was poking around on their site and found a lot of holes. I really just waltzed in. This must have been 2010 or 2011." While he was probably making this up, it's entirely plausible. **HACKED**
- Alleged infiltration of Cryptsy **HACKED**
- 2016 Bitfinex heist (~100k BTC)

Bitcoin exchanges

- Quote from a colleague about MtGox- "Yeah, I was poking around on their site and found a lot of holes. I really just waltzed in. This must have been 2010 or 2011." While he was probably making this up, it's entirely plausible. **HACKED**
- Alleged infiltration of Cryptsy **HACKED**
- 2016 Bitfinex heist (~100k BTC) **HACKED**

Bitcoin exchanges

- Quote from a colleague about MtGox- "Yeah, I was poking around on their site and found a lot of holes. I really just waltzed in. This must have been 2010 or 2011." While he was probably making this up, it's entirely plausible. **HACKED**
- Alleged infiltration of Cryptsy **HACKED**
- 2016 Bitfinex heist (~100k BTC) **HACKED**
- Dogewallet xmas hack (not an exchange)

Bitcoin exchanges

- Quote from a colleague about MtGox- "Yeah, I was poking around on their site and found a lot of holes. I really just waltzed in. This must have been 2010 or 2011." While he was probably making this up, it's entirely plausible. **HACKED**
- Alleged infiltration of Cryptsy **HACKED**
- 2016 Bitfinex heist (~100k BTC) **HACKED**
- Dogewallet xmas hack (not an exchange) **HACKED**

Bitcoin exchanges

- Quote from a colleague about MtGox- "Yeah, I was poking around on their site and found a lot of holes. I really just waltzed in. This must have been 2010 or 2011." While he was probably making this up, it's entirely plausible. **HACKED**
- Alleged infiltration of Cryptsy **HACKED**
- 2016 Bitfinex heist (~100k BTC) **HACKED**
- Dogewallet xmas hack (not an exchange) **HACKED**

Lessons?

The left-pad incident

- left-pad developer took down the left-pad package on npmjs.org, in response to NPM revoking his ownership of a repository over (basically) a trademark
- Almost everything in nodejs land somehow depends on left-pad
- If you don't remember, everything broke. The whole web broke.
- The security issue arises from the ensuing chaos: as new packages are used to replace old software, there is increased opportunity for malicious code to proliferate.
- When you ask everyone to "jump" all at the same time, where exactly will they land?

https://www.businessinsider.com/npm-left-pad-controversy-explained-2016-3

The event-stream / Bitpay Copay incident

- event-stream was a widely popular javascript library
- event-stream maintainer lost interest and someone asked to take over the library, so ownership was transferred to a new maintainer
- Then, the event-stream library was updated to use flatmap-stream
- flatmap-stream contained encrypted malicious code
- On desktops, encrypted code can be tricky- decryption keys can be based on the target's environment variables!

https://www.theblockcrypto.com/2018/11/26/bitpay-wallet-vulnerability-caused-by-use-of-popular-javascript-library/

https://www.synopsys.com/blogs/software-security/malicious-dependency-supply-chain/

bcoin removed all external dependencies in response -
https://github.com/bcoin-org/bcoin/issues/619

Information disclosure vulnerabilities

- Often overlooked
- Certain information can be inferred by the presence of other information
- HTTP 403s instead of 404s indicate something is there.
- Example: Coinbase and market effects from revealing certain cryptocurrency integration with an exchange before an announcement is made.
  - Raised questions about frontrunning and related regulatory concerns

Insufficient logging

In my experience, I have found that many web apps:

- Don't have sufficient logging for HTTP requests
- Don't monitor or review shell logs (like .bash_history etc)
- Don't configure apparmor or other systems
- Really just have no knowledge of any intrusions until something obvious happens. If an attacker is polite and cleans up after himself, the target never knows anything happened.

UTXO monitoring

I have found that web apps often don't do sufficient UTXO monitoring.

- UTXO monitoring is a simple application and should be running in multiple separate cloud systems.
- Double check that the system is running regularly
- Test the alerts and notifications for stolen coins
- Test against both testnet and regtest

Too many incidents to go over, really..

- Some JIT issues with Safari 6
  - https://bitcointalk.org/index.php?topic=416324.10
  - https://github.com/pointbiz/bitaddress.org/issues/56
- Mnemonic exfiltration using Google Analytics
- Blockchain.info RNG issues
- Incident - Nonce reuse bug was present for a few weeks
  - Unintended versions of js library was being served by cache
  - Private keys were vulnerable
- An exchange was hacked by simply asking the data center for access

Browser extensions...

- Logging into your phone carrier, there's sometimes a checkbox to disable the "extra security" feature... so be careful what browser you use when logging into your phone's account.
- So many browser extensions have contained cryptocurrency malware
- https://cryptojackingtest.com/
- https://usa.kaspersky.com/blog/razy-trojan-cryptocurrency-stealer/17048/

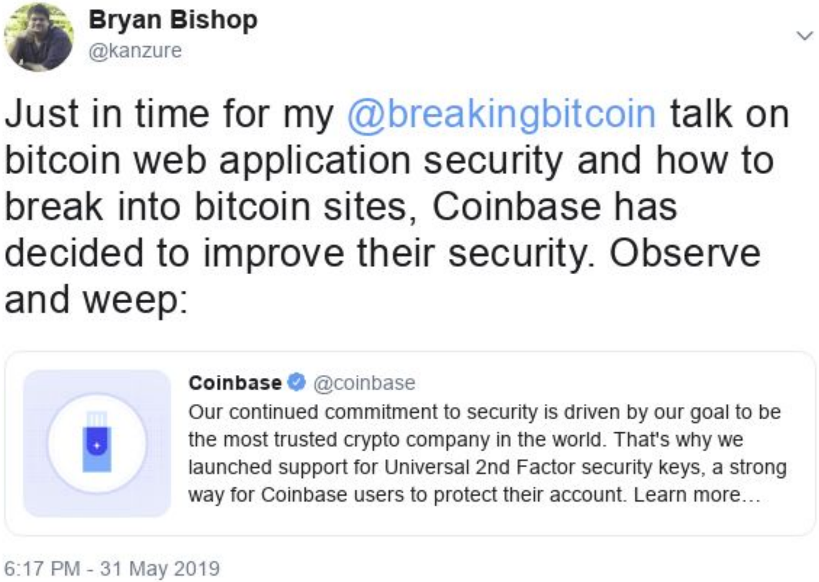
Account recovery attacks, phone numbers and 2FA

- Non-random password reset tokens
- Secret questions that really aren't secret
- Sending passwords in cleartext by email (ARRRGHH)
- SIM porting attacks (phone-based recovery is insecure)
- Some sites simply let an attacker disable 2FA by clicking a button without 2FA

WebAuthn

- Standard for web browsers to support hardware security keys; Multi-factor authentication (MFA)
- Better than generator apps (Authy...) that can be broken by restoring TOTP secrets after breaking cell phone provider security
- "Google has not had any of its 85,000+ employees successfully phished on their work-related accounts since early 2017, when it began requiring all employees to use physical security keys in place of passwords and one-time codes."
- Gemini - https://medium.com/gemini/securing-your-gemini-account-with-webauthn-b5f369b8beec
- And just announced days before this talk, Coinbase has implemented universal 2nd factor (U2F) as well - https://blog.coinbase.com/securing-your-crypto-with-security-keys-and-webauthn-551124b72d8e

Due to unforeseen circumstances, the live hacking demo has been canceled.



Bryan Bishop
@kanzure

Just in time for my @breakingbitcoin talk on bitcoin web application security and how to break into bitcoin sites, Coinbase has decided to improve their security. Observe and weep:

Coinbase @coinbase
Our continued commitment to security is driven by our goal to be the most trusted crypto company in the world. That's why we launched support for Universal 2nd Factor security keys, a strong way for Coinbase users to protect their account. Learn more...

6:17 PM - 31 May 2019

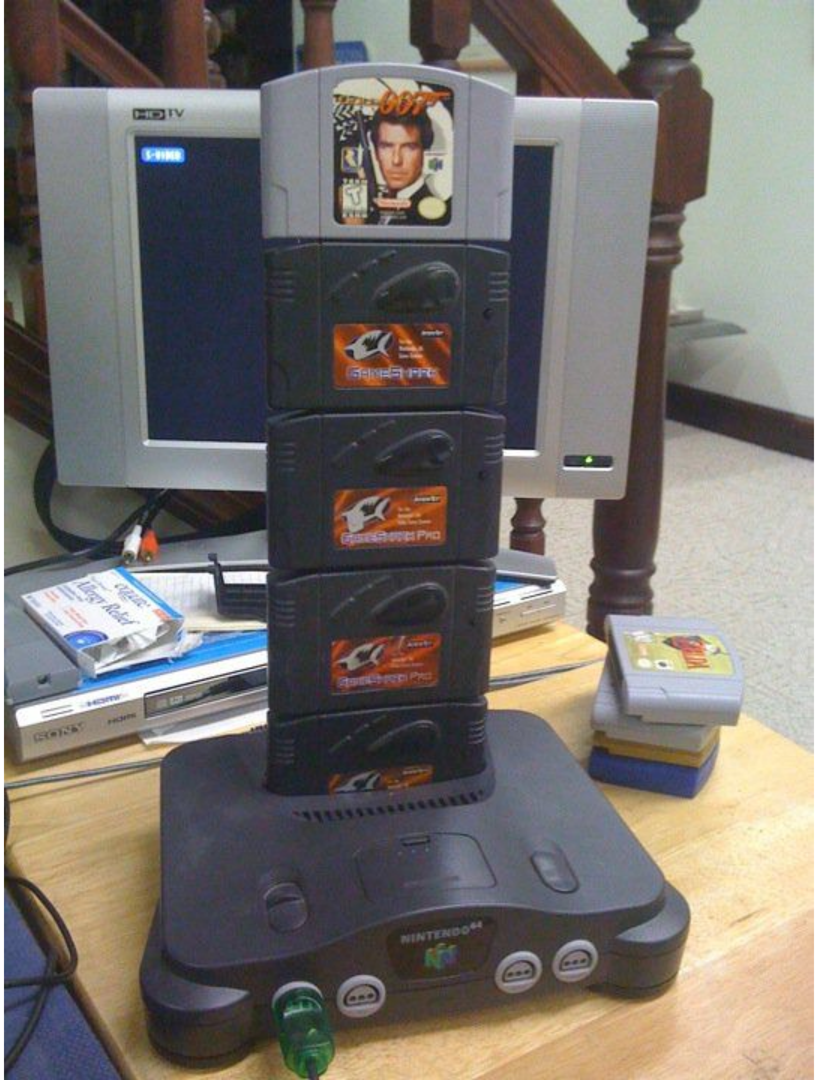
"Responsible disclosure"

- Look for companies with policies that are favorable to responsible disclosure
- Bug bounty programs
- Try to be responsible and not release zero-days if you can help it.
- Problems with "responsible disclosure" in Bitcoin Core (well, not really Bitcoin Core)

But if you really want to get good: Gaming

- Gaming provides a fun, creative environment for developing bug hunting skills, finding and playing around with glitches, etc.
- Also, there is the question of legality, which would favor game hacking over website infiltration.
- Online speedrunning communities live off of bugs and glitches that users haphazardly post to youtube, which they then convert into shorter routes.

Some videos: "Dev's Play" psychonauts, 0.5x A press video, SGDQ Pokemon Blue blindfolded glitch race


HDTV
S-VIDEO
007
GameShark Pro
GameShark Pro
GameShark Pro
NINTENDO64
SONY
HDMI
Allergy Relief

Gaming: Reverse engineering

- Games are also good targets for reverse engineering
- Example: pokered/pokecrystal
- Started in 2012
- Commented source code for Pokemon games, compiles back to original ROMs byte-for-byte for validation of correctness
- This is now the 2nd largest reverse engineering project in video games
    - Behind.... Sonic.
    - Yeah the Sonic reverse engineering community is strangely well-organized and efficient.

Takeaways

- Disable javascript
- Disable phone number/SMS recovery options on any accounts
- Don't install browser extensions
- Webapps aren't part of the bitcoin protocol
- Not your keys, not your coins
- **Don't trust, verify.**

That's all, folks.

Bryan Bishop <kanzure@gmail.com>

https://twitter.com/kanzure

0E4C A12B E16B E691 56F5 40C9 984F 10CC 7716 9FD2