# *Echavarria, et al. v. Facebook, Inc.*

3:18-cv-05982-WHA

(and all related cases)

Plaintiffs' Tutorial

January 9, 2019

# Introduction

- Plaintiffs' Counsel
  - Morgan & Morgan, P.A. Complex Litigation Group
  - Cohen Milstein Sellers & Toll PLLC
  - Milberg Tadler Phillips Grossman LLP

- Retained Experts
  - Mary T. Frantz will discuss:
    - PII, its Value, and Basic Security Against Hacking
  - Matt. B. Strebe will discuss:
    - Authentication, Access Tokens, and Hacking Tokens

- Some questions moving forward

# PII, Its Value, and Basic Security Against Hacking

- Mary Frantz
  - Over 28 years experience in cyber security, corporate enterprise technology architecture, identity and access management
  - CEO of Enterprise Knowledge Partners, LLC
  - Has served as an expert witness in several data breach cases
  - Certified Ethical Hacker, Penetration Tester, Information Systems Auditor
  - CV provided for the Court and counsel

# Personally Identifiable Information (PII)

- General Definition and California CalOPPA
  - PII Defined (California Online Privacy Protection Act (CalOPPA) Cal. Bus. & Prof. Code Sec. 2577(a))
    - Details collected on the Internet about an individual consumer, including an individual's first and last name, a physical street address, an email address, a telephone number, a Social Security number, or any other information that permits a specific individual to be contacted physically or online.

- Compromised PII has two major types
  - Temporary or "changeable" information – short shelf life
    - Examples: passwords, credit card numbers, bank accounts, drivers license, email, phone numbers
  - Historical or "unchangeable" information – long or infinite shelf life
    - Examples: original images/photos, passport numbers, current and previous addresses, mothers maiden name, relationships (family, contacts, challenge response questions), education, birth date, SSN, employment history, earnings and net worth, health history, purchase history, product designs, online comments, signed docs

# PII – Value

- Aggregated Profiles or Fullz
  - Fullz – complete “packages” of information
    - Combination of historical and temporary information
  - Highest value – Fullz aggregated behavioral and personality profile descriptors
    - Opinions, contacts, family members, style and event choices, online and physical locations visited, interests (for example: music, movies, colors, auto purchases, and sites visited),“changeable” and “unchangeable” information
    - Confidential corporate information and IP, customer complaints, confidential electronic communication
- Value of PII
  - Validated and/or recently updated Fullz PII = higher street price per profile
  - Fullz is highly coveted by nation states, “phishers,” malicious hackers, and spammers
  - Neural-marketing: the process of mining Fullz for targeted influence and manipulation
  - Collection and mining of Fullz has been used by nation state clandestine operations

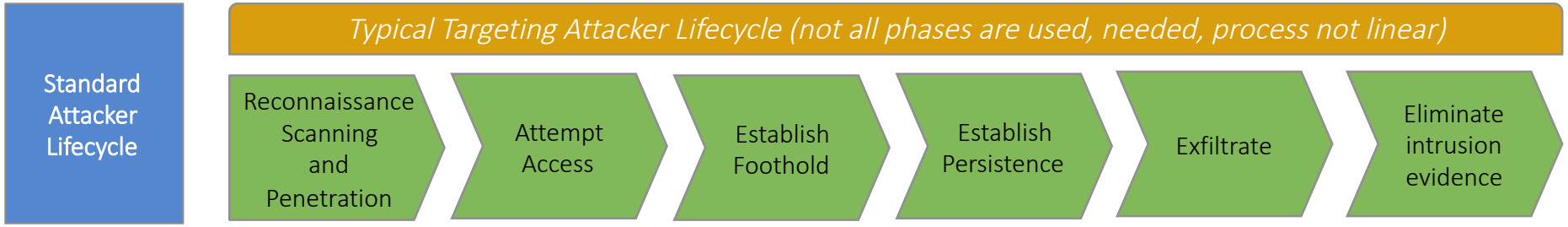
# PII – Darknet and Dark Web

- Darknet
  - System of routers and relay of devices that are not indexed or directly accessible
  - All communication between the relays use encryption
  - Cannot access Darknet using standard internet browsers; must know exact address or use Dark Web browsers
- Dark Web
  - A subset of the Darknet that works over HTML
  - Web services and specific browsers required to access Darknet
  - Built upon anonymous browsing
  - Specific, anonymous services available: messaging, emails, files sharing sites

# PII – How It is Misused

- Big data repositories created from compromised and legitimate PII on Darknet
  - Combined and blended as needed
  - Sold off in pieces for continued revenue streams
  - Pieces sold rarely equate to exact copies of stolen data (obfuscates source and trail)
  - Complete Fullz NOT usually found or sold on Darknet auction sites
- Use of Dark Web for selling stolen data
  - “Changeable” information often sold on sales or auction sites; currency is BTC
  - Pieces of Fullz are sold on sales or auction sites
    - Usually copies (not original), with diminished value
  - Complete Fullz sales use private messaging, secure emails, burner phones

# High Level Hacker Lifecycle



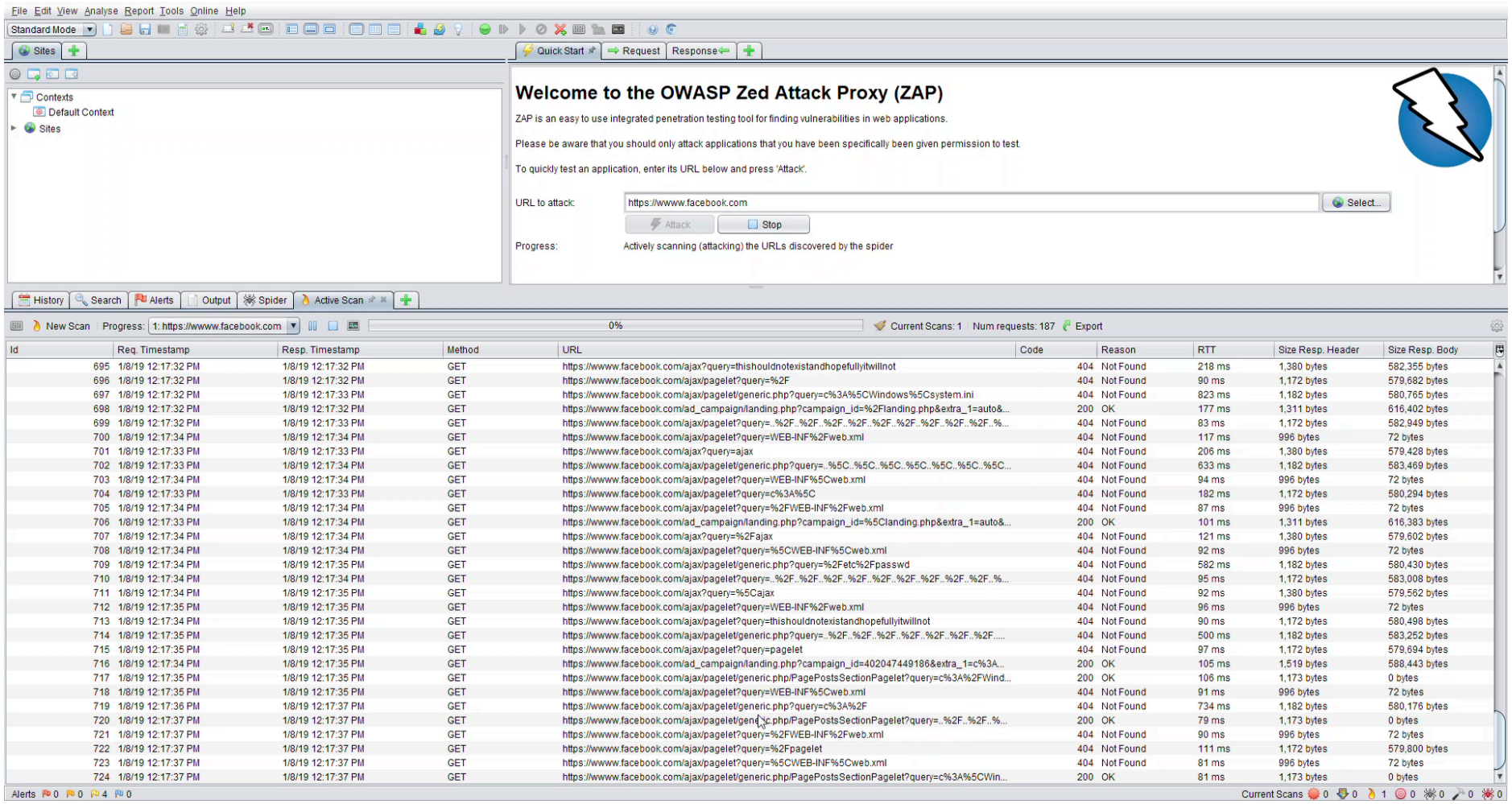

- Perform physical, logical reconnaissance (web site, employees, physical sites, etc.)
- Use web crawlers/spiders, NMAP
- Vulnerability scan
- Register as developer and gain insights
- Test possible vulnerabilities, learn from error messages/responses
- Test access – see what works, what doesn't and why

- Access and create "backdoors"
- Open up ports
- Harvest and/or elevate credentials (impersonate real user and service accounts)
- Root
- Delete evidence as they go (advanced), time stomp
- Exfiltrate

- Leave back doors
- Delete logs
- May sell or share vulnerability to script kiddies to cover tracks, create noise (cause chaos)
- Come back after noise calms down or use noise as a cover
- Watch for reactions, method of remediation

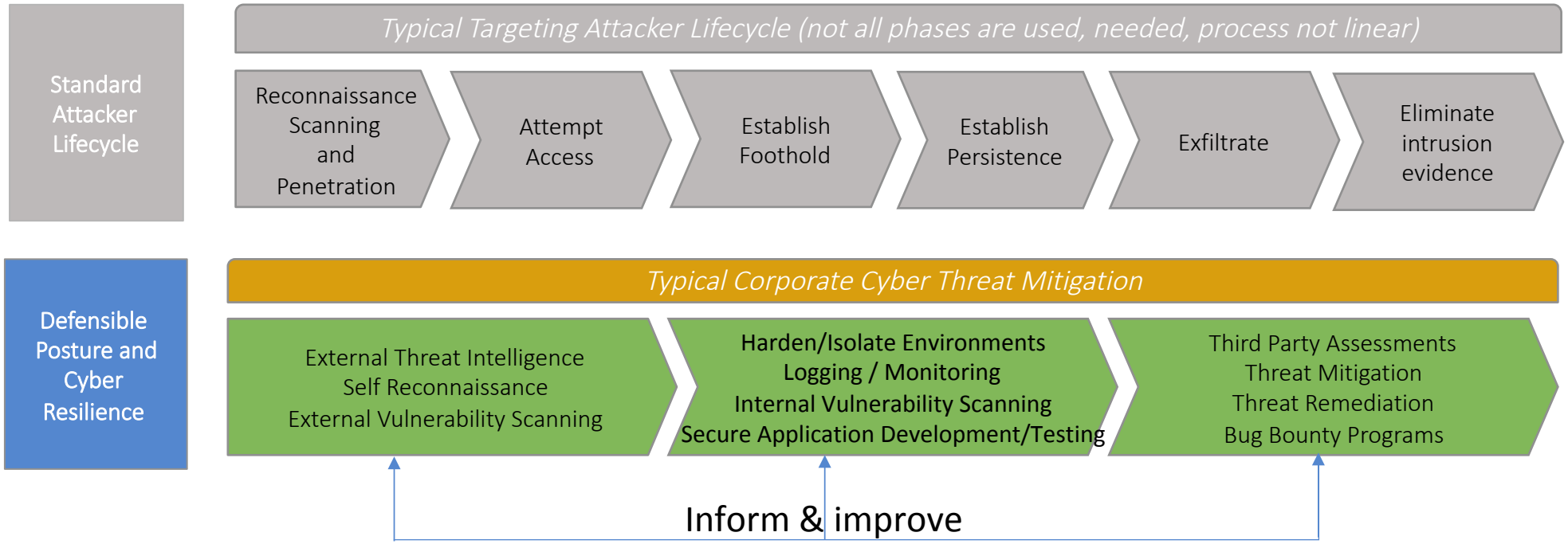
- Advanced attackers do not want to get caught, maximize "time on target"
- "Scriptkiddies" are not advanced, they often make noise that is [sometimes] easily detected

# Sample Crawl or Spider




- Using a free tool provided by OWASP called “Zap”
- Finds / crawls all URLs and calls being used by application during each step of a process
- Look for exposed information including cookies, tokens, login credentials, infrastructure info, etc.

# High Level Security Lifecycle




- Corporate Owners
- CISO, Compliance Officer, General Counsel
- Security Architecture
- SOC

- Emulate attacker/black hat reconnaissance scanning and pen testing
- Monitor the darknet for chatter about attacks, data dumps

- Monitor and log for anomalous behavior
- Constant vulnerability testing using latest signatures
- Security test/code review new releases, regression test existing software/code services/web/data stores/ apps/containers
- Harden devices/services
- Constant training/updating

- Periodic live testing of IR plan
- Security Posture Assessments
- Red Team/ Blue Team exercises
- Third party audits, internal audits
- Third party testing via bug bounty

# Securing Databases

- Generally, two types of databases – Structured and Unstructured (No-SQL)
  - No-SQL does not contain default audit trails and built-in security
  - Encrypting storage may cause unacceptable latency, but an organization can encrypt tiers based upon last access date, age of data, and other qualifiers
- Best practice No-SQL is a combination of the following:
  - Strong multi-factor authentication and authorization
  - Strong perimeter network and isolation
    - Change default ports
    - Segmentation of access rights (gateways)
    - Time-based access controls
    - Disallow concurrent access
  - Input and extract validation
  - Audit or log all plugin access (trusted and untrusted), server logs, and control access to logs
  - Replication and data segmentation, key structures
  - Strong encryption where possible

# Securing Web Apps

- OWASP (Open Web Application Security Project)
  - Globally recognized non-profit for cloud security best practices and standards
  - Industry standard:
    - Test for the Top 10 Security Risks for Cloud based applications
    - Open source vulnerability scanning software
    - Security software assurance maturity model
    - Open source tools and resources – testing tools, best practice code
  - Accepted as the standard for web-based applications for most organizations
- OWASP critical threats and penetration testing methods
  - Cross Site Scripting (XSS)
    - Enables attackers to inject client-side scripts, bypass access controls such as the same-origin policy, steal visible tokens and cookies
  - Cross Site Request Forgery (XSRF)
  - Broken Access Control

# Authentication, Access Tokens, and Hacking Tokens

- Matt Strebe
  - Over 29 years of experience in the field of cyber security, database design and security, secure network protocols, and cryptography
  - CEO of Connetic IT Services & CeNRG cloud hosting
  - Has served as an expert witness in several data breach cases
  - Author of numerous books and publications, including Network Security Foundations & Firewalls 24x7
  - Inventor, “No Transfer” (NOTX) patented device authentication protocol
  - CV provided for the Court and counsel

# Authentication

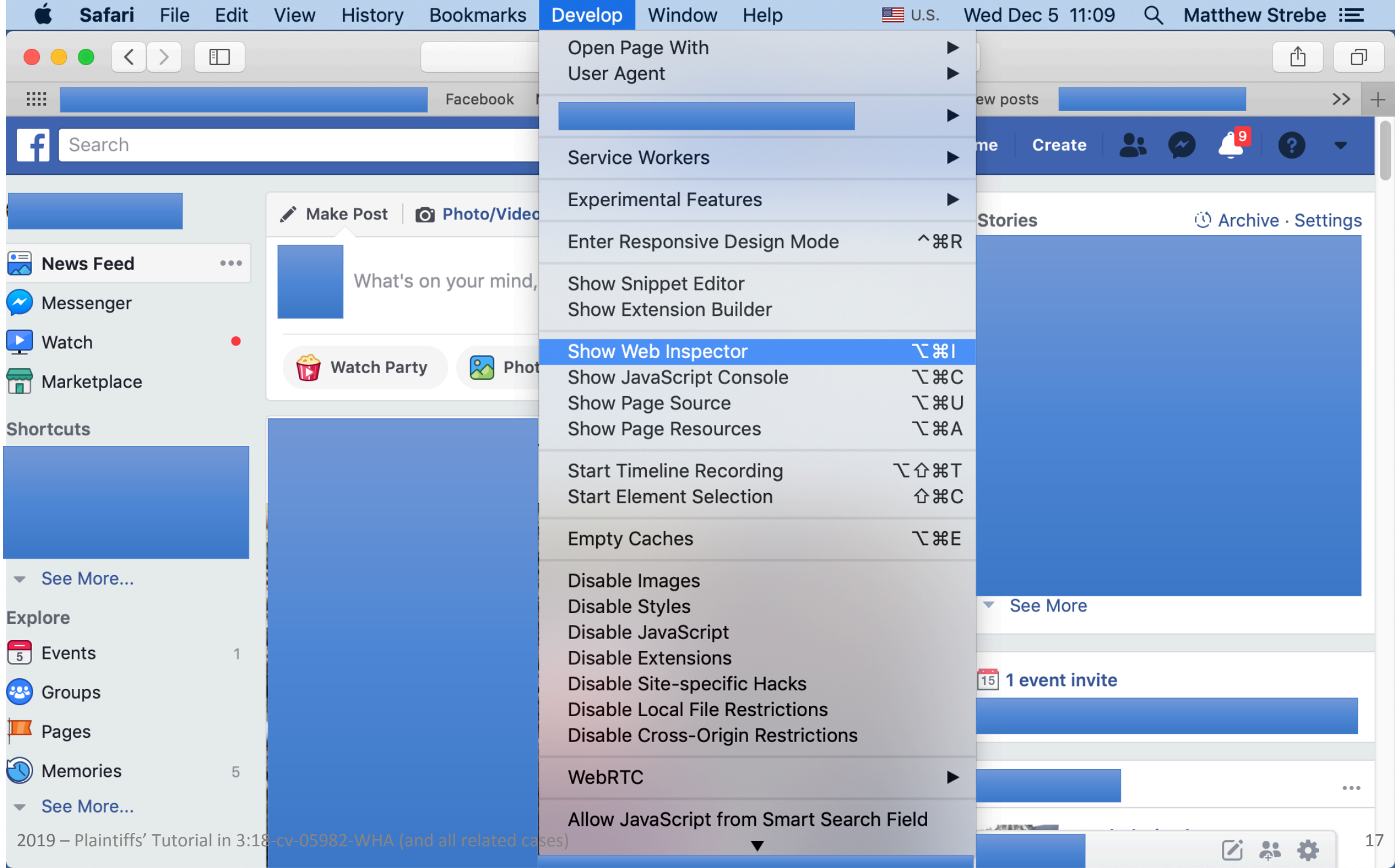
- Logging in
  - Access to Private Resources
  - Access Control Lists
- Sessions and Web Sessions
  - Website simultaneous access to multiple resources in a distributed web application
  - Authentication and Authorization are different matters
  - Here, it appears authentication is the issue, not encryption
  - Large scale web applications typically use access tokens to solve distributed access control issues

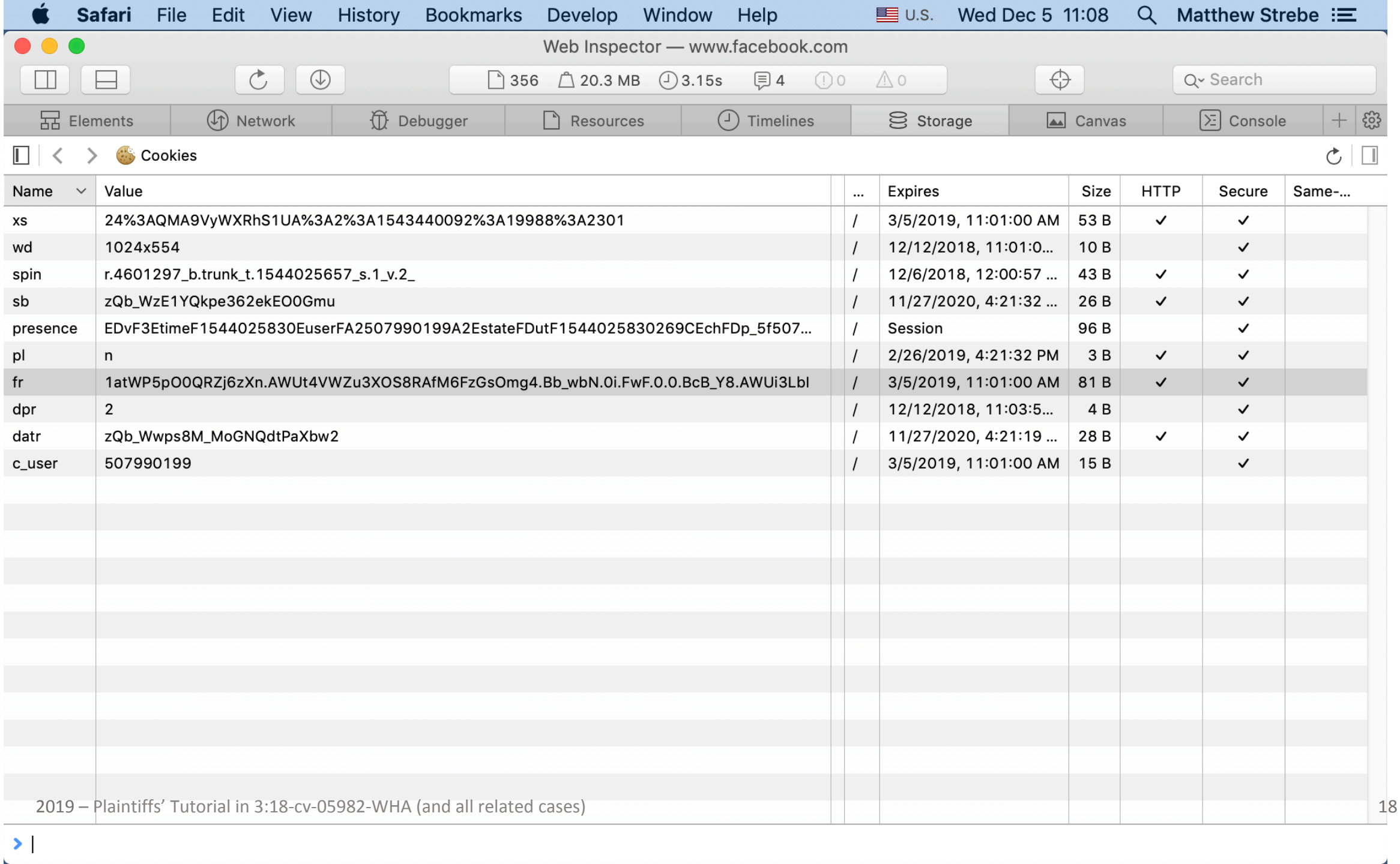
# Access Tokens

- Access Token
  - An access token is an object encapsulating the security identity of a process such as a web session. A token is used to make security decisions and to store tamper-proof information about some system entity. An access token is generated by by the logon service when a user logs on to the system
  - Bearer instrument example: court access keycards
  - Can contain anything the developer wants
  - Access Token could be limited to a single purpose (master key v. bathroom key)
- Types of Tokens
  - User Access Token (short-term, long-term)
  - App Access Token
  - Refresh Token
- For Facebook
  - Here, it appears the token Facebook associated with the “View As” function gave the hacker the same access as the original user (e.g., keycard)

# How Tokens are Transacted and Used

- Token portability (and theftability)
  - They are protected
    - On Web servers by encryption at rest
    - In transit over the Internet by encryption in flight
    - On Web browsers by encryption at rest
  - **They are not necessarily protected in the running web browser**
- Developers must be careful when sending Access Tokens to the web browser client, such that it applies only to that user
  - For developer, a conscious trade-off between security and ease-of-use
  - Expedited access requires constant vigilance upon implementation
  - Access token should never be exposed to any other users in a running browser


Safari File Edit View History Bookmarks Develop Window Help U.S. Wed Dec 5 11:09 Matthew Strebe
Open Page With
User Agent
Service Workers
Experimental Features
Enter Responsive Design Mode ^⌘R
Show Snippet Editor
Show Extension Builder
Show Web Inspector ⌥⌘I
Show JavaScript Console ⌥⌘C
Show Page Source ⌥⌘U
Show Page Resources ⌥⌘A
Start Timeline Recording ⌥⇧⌘T
Start Element Selection ⇧⌘C
Empty Caches ⌥⌘E
Disable Images
Disable Styles
Disable JavaScript
Disable Extensions
Disable Site-specific Hacks
Disable Local File Restrictions
Disable Cross-Origin Restrictions
WebRTC
Allow JavaScript from Smart Search Field
Facebook
Search
Create
Make Post
What's on your mind,
Watch Party
News Feed
Messenger
Watch
Marketplace
Shortcuts
See More...
Explore
Events 1
Groups
Pages
Memories 5
See More...
Stories
Archive · Settings
See More
1 event invite


Safari File Edit View History Bookmarks Develop Window Help
U.S. Wed Dec 5 11:08 Matthew Strebe
Web Inspector — www.facebook.com
356 20.3 MB 3.15s 4 0 0
Search
Elements Network Debugger Resources Timelines Storage Canvas Console
Cookies
Name Value ... Expires Size HTTP Secure Same-...
xs 24%3AQMA9VyWXRhS1UA%3A2%3A1543440092%3A19988%3A2301 / 3/5/2019, 11:01:00 AM 53 B ✓ ✓
wd 1024x554 / 12/12/2018, 11:01:0... 10 B ✓
spin r.4601297_b.trunk_t.1544025657_s.1_v.2_ / 12/6/2018, 12:00:57 ... 43 B ✓ ✓
sb zQb_WzE1YQkpe362ekEO0Gmu / 11/27/2020, 4:21:32 ... 26 B ✓ ✓
presence EDvF3EtimeF1544025830EuserFA2507990199A2EstateFDutF1544025830269CEchFDp_5f507... / Session 96 B ✓
pl n / 2/26/2019, 4:21:32 PM 3 B ✓ ✓
fr 1atWP5pO0QRZj6zXn.AWUt4VWZu3XOS8RAfM6FzGsOmg4.Bb_wbN.0i.FwF.0.0.BcB_Y8.AWUi3Lbl / 3/5/2019, 11:01:00 AM 81 B ✓ ✓
dpr 2 / 12/12/2018, 11:03:5... 4 B ✓
datr zQb_Wwps8M_MoGNQdtPaXbw2 / 11/27/2020, 4:21:19 ... 28 B ✓ ✓
c_user 507990199 / 3/5/2019, 11:01:00 AM 15 B ✓

# OAuth 2.0

- OAuth 2.0 is an authorization protocol frequently used as an easier authentication protocol
  - Used by companies like Facebook
  - The access token supplants other authentication steps
  - Allows third-party marketers some benefits
    - For example: simple user experience to prevent “usage walls” and encourage adoption
  - Adoption of OAuth 2.0

# OAuth 2.0 – Vulnerabilities

- But OAuth 2.0 presents a greater risk of the “bearer instrument” being misused
  - You don’t have to decrypt or comprehend an Access Token to use it
  - When you find someone else’s Access Token in a web session, you have whatever level of access that token permits within its expiration
  - Ease of coding, code re-use, complex application design, and lack of testing lead to mistakes and increased risks to users’ PII

# How Tokens Can Be Exploited

- Hackers determine that they can exploit a website to obtain another user's Access Token
- Hackers access the website as each user, access PII, then identify and steal other available Access Tokens
- And Repeat. Very quickly, hacker can obtain many millions of accounts' Access Tokens with automated scripts ("crawling")
- PII is taken and in possession of hackers and misused and/or sold

# Preliminary Questions

1. How was the “View As” token developed, security and functionality tested, prior to release?
    a) What process did Facebook use to test the “View As” feature prior to release?
    b) Where else was/is the token used?
    c) What Security Development (i.e. Secure SDLC) processes does Facebook use to test their software prior to release?
2. When did Facebook first detect the issue, and how?
3. When did Facebook identify the root cause of the issue, and how?
4. Any report done by Facebook or third-party?
5. How did Facebook determine the affected entities?
6. What steps did Facebook take to contain and remediate the issue?