Are You on Tinder?

Someone May Be Watching You Swipe

Introduction

Launched in 2012, Tinder is one of the first “swiping apps” allowing users to swipe through profiles to ultimately make social connections; swiping right for a profile they like, swiping left to move on to the next profile indicating lack of interest or “super liking” with an upward swipe.

Although Tinder’s mission* is "empowering users around the world to create new connections that otherwise might never have been possible”, the application is most commonly used as a dating platform. According to its website, Tinder has matched over 20 billion people to date, and is used in 196 countries.

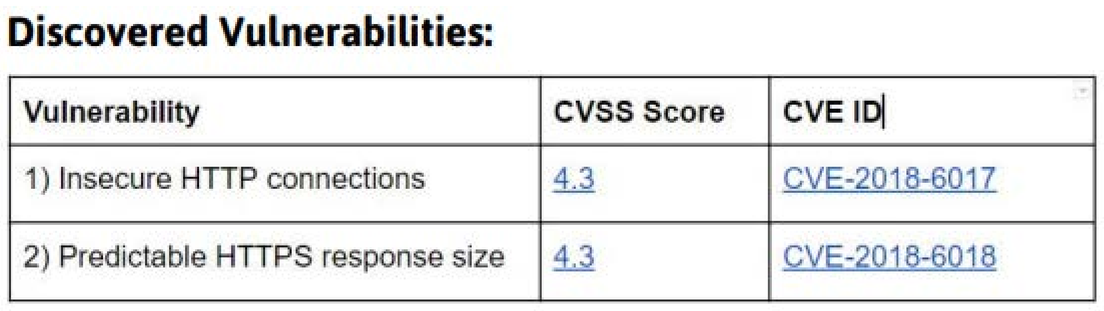
Due to its nature, Tinder users regularly upload personal information alongside photos of themselves, with the assumption that this application is secure and protective of their data. However, Checkmark’s Security Research Team recently conducted elaborate research which exposed worrisome security vulnerabilities in this highly popular application.

The team went through the responsible disclosure process, sending a full report to the Tinder security team and notifying them of our intention to publish our findings. The following paper presents the research and its findings on both the Tinder’s Android and iOS versions.

Research Findings

Our research found two vulnerabilities that, once combined, enable a malicious attacker to spy on a Tinder user’s every move in the app. This means the attacker can see the user’s profile, the profiles the user views and the actions he or she takes (for example, swiping left / right and “super liking”). The attacker can follow the user’s Tinder matches and seriously compromise the user’s privacy.

In order to carry out the attack, the attacker needs to be on the same WiFi network as the user. This is possible via any public hotspot. Other scenarios where an attacker can intercept traffic include VPN or company administrators, DNS poisoning attacks or a alicious internet service provider - to name a few.



Discovered Vulnerabilities:

Vulnerability	CVSS Score	CVE ID
1) Insecure HTTP connections	4.3	CVE-2018-6017
2) Predictable HTTPS response size	4.3	CVE-2018-6018

CVSS: https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:A/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N
CVE 6017: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-6017
CVE 6018: https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-6018

* https://www.gotinder.com/press?locale=en

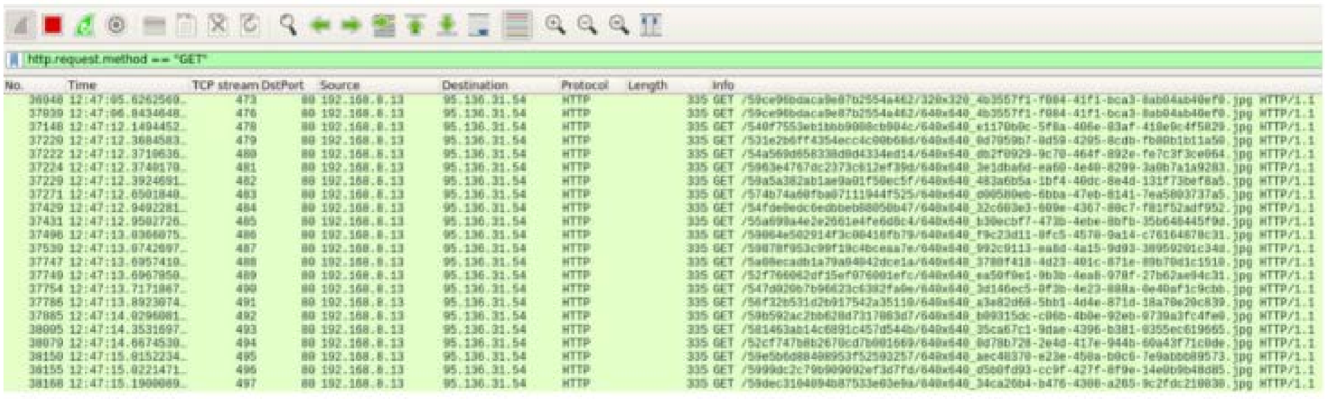
In order to demonstrate an attack, we produced a fully functional app, “TinderDrift” - any attacker could have developed this app as well. To see a full demonstration of how this app was used to follow Tinder users’ actions on Tinder, see this video: https://youtu.be/ZBTL1bmJ9o8

Use Cases and Attack Scenarios

The following are a few examples of how the vulnerabilities mentioned above can be used to perform a malicious attack.

1. Insecure HTTP Connections

Both iOS and Android versions of Tinder make insecure HTTP requests when downloading user profile pictures. The behavior is slightly different between iOS and Android but the result is similar: Attackers can easily discover what device is viewing which profiles. Furthermore, if the user stays online long enough, or if the app initializes while on the vulnerable network, the attacker can identify and explore the user’s profile



```
http.request.method == "GET"

No.   Time                 TCP stream DstPort Source        Destination   Protocol Length Info
36048 12:47:05.6262560…    473        80 192.168.8.13  95.136.31.54  HTTP     335 GET /59ce96bdaca0e87b2554a462/320x320_4b3557f1-f084-41f1-bca3-8ab04ab40ef0.jpg HTTP/1.1
37039 12:47:06.8434648…    476        80 192.168.8.13  95.136.31.54  HTTP     335 GET /59ce96bdaca0e87b2554a462/640x640_4b3557f1-f084-41f1-bca3-8ab04ab40ef0.jpg HTTP/1.1
37148 12:47:12.1494452…    478        80 192.168.8.13  95.136.31.54  HTTP     335 GET /540f7553eb1bbb9080cb904c/640x640_e1170b0c-5f8a-406e-83af-410e0c4f5829.jpg HTTP/1.1
37220 12:47:12.3684583…    479        80 192.168.8.13  95.136.31.54  HTTP     335 GET /531e2b6ff4354ecc4c00b68d/640x640_0d7059b7-0d59-4295-8cdb-fb80b1b11a50.jpg HTTP/1.1
37222 12:47:12.3710636…    480        80 192.168.8.13  95.136.31.54  HTTP     335 GET /54a560d658330d0d4334ed14/640x640_db2f0929-9c70-464f-892e-fe7c3f3ce064.jpg HTTP/1.1
37224 12:47:12.3740170…    481        80 192.168.8.13  95.136.31.54  HTTP     335 GET /5963e4767dc2373c612ef39d/640x640_3e1dbafd-ea60-4e40-8299-3a0b7a1a9283.jpg HTTP/1.1
37229 12:47:12.3924691…    482        80 192.168.8.13  95.136.31.54  HTTP     335 GET /59a5a382ab1ae9a01f50ec5f/640x640_483a6b5a-1bf4-40dc-8e4d-131f73bef8a5.jpg HTTP/1.1
37271 12:47:12.6501840…    483        80 192.168.8.13  95.136.31.54  HTTP     335 GET /574b74a60fba07111944f525/640x640_d00580eb-6bba-47eb-8141-7ea5803737a5.jpg HTTP/1.1
37429 12:47:12.9492281…    484        80 192.168.8.13  95.136.31.54  HTTP     335 GET /54fde0edc6edbbeb68050b47/640x640_32c603e3-609e-4367-80c7-f81f52adf952.jpg HTTP/1.1
37431 12:47:12.9502726…    485        80 192.168.8.13  95.136.31.54  HTTP     335 GET /55a698a4e2e2661e4fe6d8c4/640x640_b30ecbf7-473b-4ebe-8bfb-35b648445f9d.jpg HTTP/1.1
37496 12:47:13.0366075…    486        80 192.168.8.13  95.136.31.54  HTTP     335 GET /59064e502914f3c08416fb79/640x640_f9c23d11-0fc5-4570-9a14-c76164670c31.jpg HTTP/1.1
37530 12:47:13.0742697…    487        80 192.168.8.13  95.136.31.54  HTTP     335 GET /59870f953c99f19c4bceaa7e/640x640_992c0113-ea8d-4a15-9d93-38959201c348.jpg HTTP/1.1
37747 12:47:13.6957410…    488        80 192.168.8.13  95.136.31.54  HTTP     335 GET /5a08ecadb1a79a04042dce1a/640x640_3780f418-4d23-401c-871e-89b70d1c1510.jpg HTTP/1.1
37749 12:47:13.6967950…    489        80 192.168.8.13  95.136.31.54  HTTP     335 GET /52f766062df15ef076001efc/640x640_ea50f0e1-9b3b-4ea8-970f-27b62ae94c31.jpg HTTP/1.1
37754 12:47:13.7171867…    490        80 192.168.8.13  95.136.31.54  HTTP     335 GET /547d020b7b96623c6382fa0e/640x640_3d146ec5-0f3b-4e23-888a-0e40af1c9cbb.jpg HTTP/1.1
37786 12:47:13.8923074…    491        80 192.168.8.13  95.136.31.54  HTTP     335 GET /56f32b531d2b917542a35110/640x640_a3e82d68-5bb1-4d4e-871d-18a70e20c839.jpg HTTP/1.1
37885 12:47:14.0296001…    492        80 192.168.8.13  95.136.31.54  HTTP     335 GET /59b592ac2bb628d7317083d7/640x640_b09315dc-c06b-4b0e-92eb-0739a3fc4fe0.jpg HTTP/1.1
38005 12:47:14.3531697…    493        80 192.168.8.13  95.136.31.54  HTTP     335 GET /581463ab14c6891c457d544b/640x640_35ca67c1-9dae-4396-b381-0355ec619665.jpg HTTP/1.1
38079 12:47:14.6674530…    494        80 192.168.8.13  95.136.31.54  HTTP     335 GET /52cf747b8b2670cd7b001669/640x640_0d78b728-2e4d-417e-944b-60a43f71c0de.jpg HTTP/1.1
38150 12:47:15.0152234…    495        80 192.168.8.13  95.136.31.54  HTTP     335 GET /59e5b6d08400953f52593257/640x640_aec48370-e23e-450a-b0c6-7e9abbb89573.jpg HTTP/1.1
38155 12:47:15.0221471…    496        80 192.168.8.13  95.136.31.54  HTTP     335 GET /5999dc2c79b909092ef3d7fd/640x640_d5b0fd93-cc9f-427f-8f9e-14e0b9b48d85.jpg HTTP/1.1
38168 12:47:15.1900069…    497        80 192.168.8.13  95.136.31.54  HTTP     335 GET /59dec3104094b87533e03e9a/640x640_34ca26b4-b476-4308-a265-9c2fdc210830.jpg HTTP/1.1
```

```
GET http://images.gotinder.com/55e267589951180400fa0e8a8/640x640_20da3188-5d36-40a1-9ac0-
c8de083c0516.jpg
```

Highlighted in green is the user ID, in yellow the photo resolution and the orange is a random uuid. Requests that are made with a 640x640 photo resolution while the user is browsing the app lead to the profile currently shown on the user’s screen or about to be shown on his/her screen. Requests with a 320x320 photo resolution are connected to the user profile under attack. This is consistent in the iOS version of Tinder but only randomly happens on the Android version.

Requests with a 84x84 photo resolution mean the user just opened a chat window (a reduced version of the profile picture appears in the chat window). The above demonstrates how using HTTP allows for snooping around a victim’s profile, but it also allows an attacker to intercept and modify traffic. Profile images that the victim sees can be swapped, rogue advertising can be placed and malicious content can be injected (if, as an example, the targeted device suffers from an image format vulnerability).

Demo: https://youtu.be/azg1PIb7U3c

Recommendations:

We highly recommend switching HTTP to HTTPS. One might argue that this affects speed quality, but when it comes to the privacy and sensitivity needed, speed should not be the main concern.

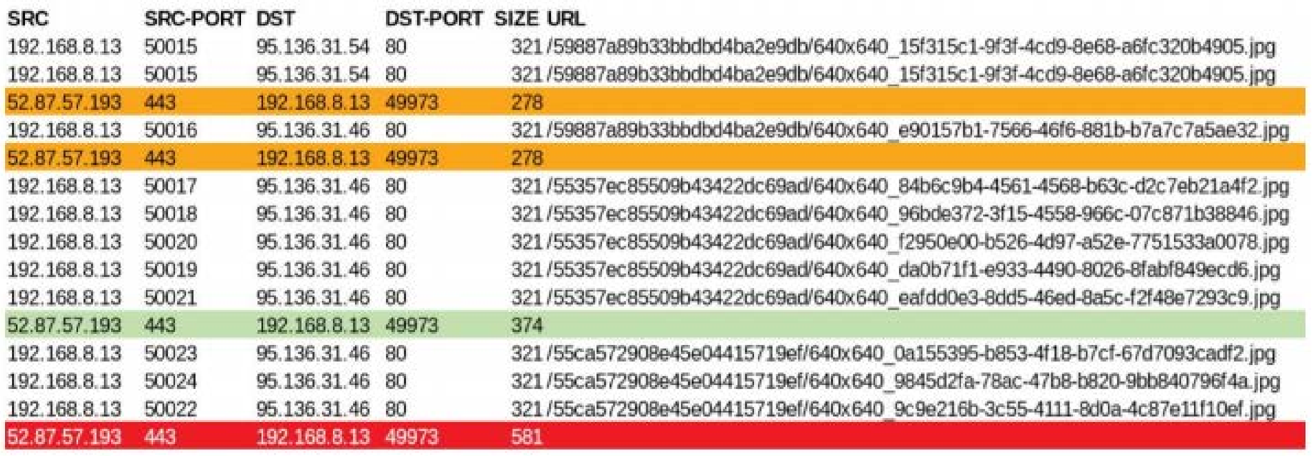
2. Predictable HTTPS Response Size

By carefully analyzing the traffic coming from the client to the API server and correlating with the HTTP image requests traffic, it is possible for an attacker to determine not only which image the user is seeing on Tinder, but also which action did the user take. This is done by checking the API server's encrypted response payload size to determine the action.

This is how the traffic flow works in a nutshell:

- Tinder is initialized and may (or may not) request a profile image from http://images.gotinder.com
- The app then makes many requests from the main profile photo to cache in the device from http://images.gotinder.com
- The app requests the secondary photos of the next profile to be shown (to speed things up) from http://images.gotinder.com
- User input is required, swipe left, right, or up
- A request is made and a response is received from api.gotinder.com. The response size changes for each action
- If main photo cache is empty, the app can then go to (2) otherwise to (3)

By correlating the response payload size in (5) with the secondary image request in (3), an attacker can conclude which action the user took for each profile he/she went through.



SRC	SRC-PORT	DST	DST-PORT	SIZE	URL
192.168.8.13	50015	95.136.31.54	80	321	/59887a89b33bbdbd4ba2e9db/640x640_15f315c1-9f3f-4cd9-8e68-a6fc320b4905.jpg
192.168.8.13	50015	95.136.31.54	80	321	/59887a89b33bbdbd4ba2e9db/640x640_15f315c1-9f3f-4cd9-8e68-a6fc320b4905.jpg
52.87.57.193	443	192.168.8.13	49973	278	
192.168.8.13	50016	95.136.31.46	80	321	/59887a89b33bbdbd4ba2e9db/640x640_e90157b1-7566-46f6-881b-b7a7c7a5ae32.jpg
52.87.57.193	443	192.168.8.13	49973	278	
192.168.8.13	50017	95.136.31.46	80	321	/55357ec85509b43422dc69ad/640x640_84b6c9b4-4561-4568-b63c-d2c7eb21a4f2.jpg
192.168.8.13	50018	95.136.31.46	80	321	/55357ec85509b43422dc69ad/640x640_96bde372-3f15-4558-966c-07c871b38846.jpg
192.168.8.13	50020	95.136.31.46	80	321	/55357ec85509b43422dc69ad/640x640_f2950e00-b526-4d97-a52e-7751533a0078.jpg
192.168.8.13	50019	95.136.31.46	80	321	/55357ec85509b43422dc69ad/640x640_da0b71f1-e933-4490-8026-8fabf849ecd6.jpg
192.168.8.13	50021	95.136.31.46	80	321	/55357ec85509b43422dc69ad/640x640_eafdd0e3-8dd5-46ed-8a5c-f2f48e7293c9.jpg
52.87.57.193	443	192.168.8.13	49973	374	
192.168.8.13	50023	95.136.31.46	80	321	/55ca572908e45e04415719ef/640x640_0a155395-b853-4f18-b7cf-67d7093cadf2.jpg
192.168.8.13	50024	95.136.31.46	80	321	/55ca572908e45e04415719ef/640x640_9845d2fa-78ac-47b8-b820-9bb840796f4a.jpg
192.168.8.13	50022	95.136.31.46	80	321	/55ca572908e45e04415719ef/640x640_9c9e216b-3c55-4111-8d0a-4c87e11f10ef.jpg
52.87.57.193	443	192.168.8.13	49973	581	

The packets are sent from the Tinder app to the image server on port 80 (HTTP) and replies from the API server on port 443 (HTTPS) after the user completed an action. The first case, highlighted in orange, represents a swipe left (meaning "moving on to the next profile"). First, the app requests two additional profile photos via HTTP, then the user takes another action and the API server replies with a 278 byte encrypted response.The second case, also highlighted orange, is similar however there was only one previous HTTP request - again, 278 bytes for a left swipe.

In the third case, highlighted green, we see five previous additional profile image requests, followed by a 374 byte response, indicating that the user has swiped right. Finally, the forth case, highlighted in red, shows a "liked" and "matched" response, with 581 bytes, that followed three secondary profile HTTP requests. This way, by carefully inspecting network traffic and monitoring what the user is doing, an attacker can pinpoint each of the user's actions.

It is important to note that while recently analyzing the Android app, the research team noticed that the latest Tinder version has certificate pinning in place. This makes it harder to analyze the HTTPS traffic to the API server (api.gotinder.com) even when a rogue certificate authority has been installed on the device.

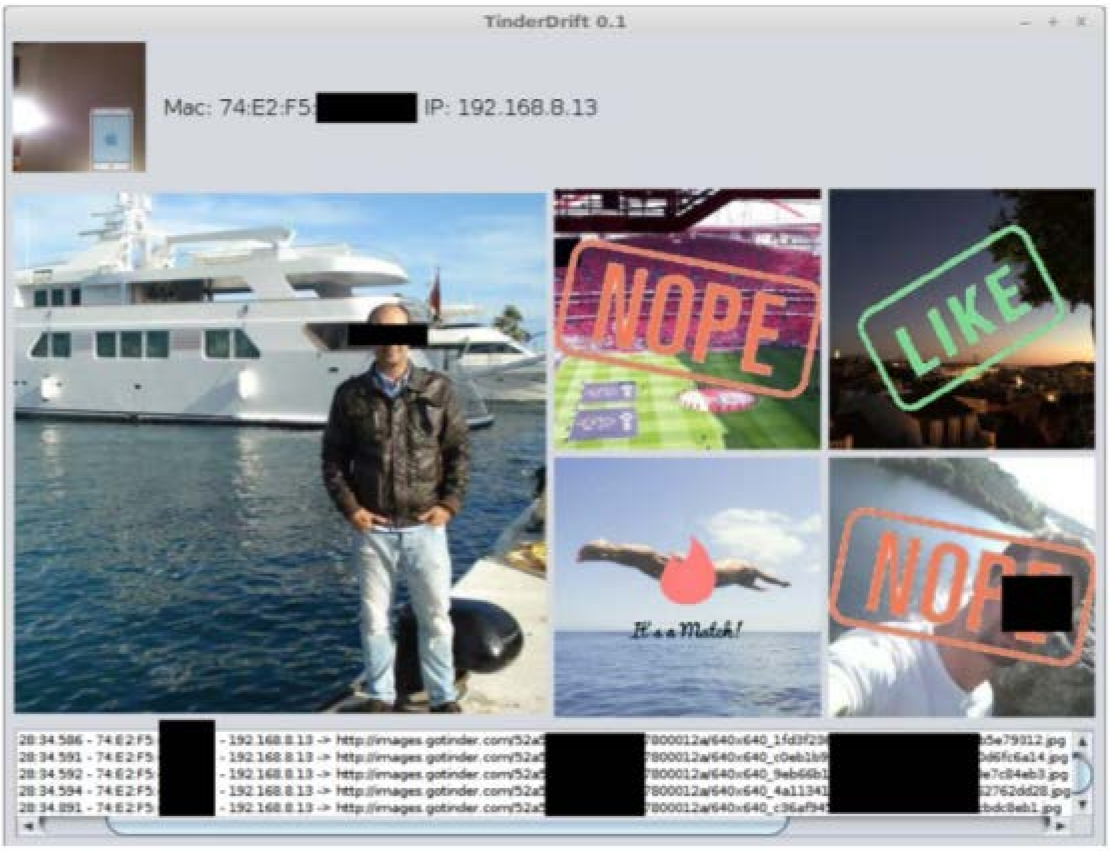
This is good news for the app's Android users, given that communications to the API server are properly secured via HTTPS, an attacker cannot simply snoop the traffic to analyze what the Tinder app is sending to the server since it is encrypted.

Recommendations:

User responses should not be predictable. Padding the requests and responses should be considered in order to reduce the information available to an attacker. If the responses were padded to a fixed size, it would be impossible to differentiate between them. Otherwise, even those encrypted, the responses contain valuable information.

3. Combining the Two Vulnerabilities

An attacker located in a public space with available WiFi (an airport, restaurant, etc.) can log all HTTP traffic going to Tinder servers and analyze Tinder's API server responses. This means the attacker can collect information about all of the app's users on the same WiFi network. The attacker can follow how users are using the app in real time. This attack method enables an attacker to blackmail vulnerable users; for example, consider those users cheating on their significant others, users uploading sensitive images, users declaring sensitive sexual information, and so on.



Mac: 74:E2:F5:[redacted] IP: 192.168.8.13

```
28:34.586 - 74:E2:F5:[redacted] - 192.168.8.13 -> http://images.gotinder.com/52a[redacted]7800012a/640x640_1fd3f29[redacted]b5e79312.jpg
28:34.591 - 74:E2:F5:[redacted] - 192.168.8.13 -> http://images.gotinder.com/52a[redacted]7800012a/640x640_c0eb1b9[redacted]0d6fc6a14.jpg
28:34.592 - 74:E2:F5:[redacted] - 192.168.8.13 -> http://images.gotinder.com/52a[redacted]7800012a/640x640_9eb66b1[redacted]e7c84eb3.jpg
28:34.594 - 74:E2:F5:[redacted] - 192.168.8.13 -> http://images.gotinder.com/52a[redacted]7800012a/640x640_4a11341[redacted]2762dd28.jpg
28:34.891 - 74:E2:F5:[redacted] - 192.168.8.13 -> http://images.gotinder.com/52a[redacted]7800012a/640x640_c36af94[redacted]bdc8eb1.jpg
```

Conclusion

It is possible for an attacker to take advantage of two apparently minor and unrelated vulnerabilities in order to infer a Tinder user’s screen, completely invading the target’s privacy. The assumption that HTTP can be used in a sensitive application must be dropped.

Standard HTTP is vulnerable to eavesdropping and content modification, introducing potential threats that might not even be related to the app itself but the underlying operating system and/or used libraries.

HTTPS increases security overall and nowadays seems to be even faster than HTTP in most cases. The use of HTTP allows for the escalation of other types of attacks, such as the Response Size Predictability, shown in this paper.

We highly recommend our readers to be mindful of the likelihood of such attacks on their privacy and to avoid public networks when possible, as these are highly vulnerable.