

Mobile Dating Vulnerabilities: Exploitation Through Social Engineering

Author: David Bernstein

Advisor: Ming Chow

Submission Date: December 14, 2016

Abstract:

According to MITRE's Common Attack Pattern Enumeration and Classification (CAPEC) list, social engineering (ID 403) is defined as "The manipulation and exploitation of people. The techniques defined by each pattern are used to convince someone into performing actions or divulging confidential information, often resulting in access to computer systems or facilities" [1]. Since the rise of new mobile dating applications, social engineering has been a hot topic of debate because it relies on the inherent, flawed nature of human beings rather than flawed code. One can easily conduct a social engineering experiment in order to get users on these applications to divulge sensitive, private information simply by posing as a desirable companion. This paper examines this vulnerability and addresses how one can defend against it by using both scripts and a healthy dose of skepticism.

Introduction:

Online dating has been around since 1997 with the website *sixdegrees.com* [2]. Since then, websites such as BlackPlanet, MiGente, Friendster, and more have all been avenues for online dating. Currently, however, these applications have become even more prevalent due to the fact that they are on mobile devices, which are used on a far more frequent basis than a website due to convenience and accessibility. Applications such as Tinder, JSwipe, Bumble, and Hinge have all been growing in popularity recently, with some studies showing download rates of 20,000 downloads per day [3]. In fact, studies show that one in ten Americans use these applications [3]. These new applications display vast amounts of user information to other users, such as school, workplace, sexual orientation, age, and even a general location of a user based on

proximity [4]. Users are looking for a wide variety of outcomes, ranging from committed relationships to simple sexual encounters [5].

Naturally, location-based applications that allow users to place themselves in semi-vulnerable positions are security risks, with investigations constantly happening [6]. Instead of taking a technical approach to the securities issues, however, one can take a social engineering perspective. This paper will prove that these applications are inherently a security risk because users are far too trusting with their data. By creating a Tinder account that automatically matches and messages users, this paper will prove this security flaw and will be present the response of users a proof. This automatic user was created using the code from <https://github.com/alkawryk/tinderbot>, which is publically available. The code was only altered such that it sent the message “*Hey, want to meet up tonight? Where are ya?*” to everyone it matched to.

To The Community

I found this topic to be incredibly prevalent due to the incredibly high number of young adults who use dating applications. Most users use the applications carelessly, meaning that they have no regard for security or safety. This is because the prospect of being physically validated, finding a meaningful connection, or whatever else people are looking for is, in some people’s opinion, worth forsaking their security for.

This fact I find deeply troubling. People will willingly give out their phone numbers or other personal information. They will even invite strangers to their own domicile without thought as to whether or not this person should be trusted.

After creating a fake account, I was practically blindsided by how *easy* it is! One can make a fake Facebook and Tinder account literally within minutes. You can steal pictures of anyone online by simply downloading them, and then uploading them as your own. Shockingly, it personally took me only a couple minutes of online research before I was able to quickly set up these accounts and begin to infiltrate these social networks through completely “legitimate” ways.

I learned from this analysis that you have to treat everything online with a healthy amount of skepticism because *anyone* can make *anything* and put it online for anyone to see. Is that person you are messaging on Tinder really interested in you? Or maybe it is just someone pretending to be someone they are not in order to get information or money from you. Caution *needs* to be exercised *constantly* on these applications.

Defenses:

The recommended defense for this type of vulnerability is a more active approach than what is used now. Currently, users are given a brief warning at the beginning of their usage of the application about not giving out private information. This warning, however, is far too passive. A more active defense mechanism would be one that would locally scan every message sent and try to find common patterns that reveal too much information about users. Information such as emails, phone numbers, and addresses should be scanned in a way that the application reminds the user that they are revealing sensitive information about themselves and should continue with caution. This way, people who are carelessly sending personal information will be made more aware of their decisions. An example of this code can be seen below:

```
# filter.py  
# David Bernstein
```

```
# Last update: 12/14/2016
# Purpose: to scan through a conversation and find any plaintext
addresses
#         in order to warn the user of a security warning.
# Usage: python filter.py
# Sources:
# https://www.codeproject.com/Tips/989012/Validate-and-Find-
Addresses-with-Regex
```

```
import re
statesList =
'Alabama|Alaska|Arizona|Arkansas|California|Colorado|Connecticut
|Delaware|Florida|Georgia|Hawaii|Idaho|Illinois|Indiana|Iowa|Kan
sas|Kentucky|Louisiana|Maine|Maryland|Massachusetts|Michigan|Min
nesota|Mississippi|Missouri|Montana|Nebraska|Nevada|New[ ]Hampshi
re|New[ ]Jersey|New[ ]Mexico|New[ ]York|North[
]Carolina|North[ ]Dakota|Ohio|Oklahoma|Oregon|Pennsylvania|Rhode[
]Island|South[ ]Carolina|South[ ]Dakota|Tennessee|Texas|Utah|Vermo
nt|Virginia|Washington|West[ ]Virginia|Wisconsin|Wyoming'
```

```
statesAbbvsList =
'AL|AK|AS|AZ|AR|CA|CO|CT|DE|DC|FM|FL|GA|GU|HI|ID|IL|IN|IA|KS|
KY|LA|ME|MH|MD|MA|MI|MN|MS|MO|MT|NE|NV|NH|NJ|NM|NY|NC|ND|MP|OH|O
K|OR|PW|PA|PR|RI|SC|SD|TN|TX|UT|VT|VI|VA|WA|WV|WI|WY'
```

```
streetRegex = '\d+[ ](?:[A-Za-z0-9.-]+[
]?)+(?:Avenue|Lane|Road|\
Boulevard|Drive|Street|Ave|Dr|Rd|Blvd|Ln|St)\.??'
```

```
emailRegex = '[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,}'
```

```
phoneRegex = '[2-9]\d{2}-\d{3}-\d{4}'
```

```
conversation = 'My number is 772-123-2134 C meet me at my place!
I live at 200 College Ave, Massachusetts (MA), 02155.
You can also email me at user@user.com. '
```

```
# Analyze conversation for streets, states, and
streets = re.findall(streetRegex, conversation)
states = re.findall(statesList, conversation)
statesAbbvs = re.findall(statesAbbvsList, conversation)
zipCodes = re.findall("\d{5}(?:[-\s]\d{4})?", conversation)
emails = re.findall(emailRegex, conversation)
phones = re.findall(phoneRegex, conversation)
```

```
if len(streets) or len(states) or len(statesAbbvs) or
len(zipCodes) or len(emails) or len(phones):
```

```
print 'SECURITY WARNING! Do not give out your address to
strangers!'
print 'Private information being given away:'
if len(streets):
    print 'Streets:'
    print streets
if len(states) or len(statesAbbvs):
    print 'States:'
    print states
    print statesAbbvs
if len(zipCodes):
    print 'Zip Codes:'
    print zipCodes
if len(emails):
    print 'Emails:'
    print emails
if len(phones):
    print "Phone Numbers:"
    print phones
```

This code will immediately search through a conversation (given, in this example, by the string “conversation”, and will pick out sensitive information that the user should be warned about before sharing. If any sensitive information is found, it will warn the user about what kind of information is being shared, and then tell the user exactly what sensitive information is being detected.

Conclusion:

The results from the Tinder bot testing were surprising due to how trusting users were. Many people were immediately willing to disclose their current location, phone number, and more just in order to meet a fictional character. The bot’s Tinder and Facebook accounts were deleted on December 14, 2016 in order to protect the users who fell victim to this trap.

One should take away from this paper the fact that fake profiles are everywhere. It is *unsettlingly easy* to create a fake account, and it can be extremely difficult to differentiate a fake account from a real account. One should use secure, 3rd party channels to communicate, and if they want to meet someone, should choose to do so in a public, open place in order to remain as

safe as possible. On the development side, one can only hope that companies take a more active stance on user safety, or else these companies are doing a major disservice to their active user base.

References:

1. N/A. *CAPEC Category: Social Engineering*. Common Attack Pattern Enumeration and Classification 2015; Available from: <https://capec.mitre.org/data/definitions/403.html>.
2. Boyd, D., Ellison, N., *Social Network Sites: Definition, History, and Scholarship*. Journal of Computer-Mediated Communication, 2007. **13**(1): p. 210-230.
3. James, J., *MOBILE DATING IN THE DIGITAL AGE: COMPUTER-MEDIATED COMMUNICATION AND RELATIONSHIP BUILDING ON TINDER*. 2015, Texas State University: Mass Communications. p. 54.
4. Choo, R., et al. *INFORMATION SYSTEMS SECURITY, ASSURANCE AND PRIVACY (SIGSEC)*. 2015; Available from: <http://aisel.aisnet.org/amcis2015/ISSecurity/GeneralPresentations/13/>.
5. Orosz, G.e.a., *Too many swipes for today: The development of the Problematic Tinder Use Scale (PTUS)*. Journal of Behavioral Addictions, 2014. **5**(3).
6. Farnden, J., Martini, B., Choo, K., *Privacy Risks in Mobile Dating Apps*. Proceedings of 21st Americas Conference on Information Systems, 2015.