Abusing Social Networks with Abuse Reports
A Coalition Attack for Social Networks

Slim Trabelsi[1] and Hana Bouafif[2]
[1]*SAP Labs France, 805, Av Dr Maurice Daunat, Mougins, France*
[2]*ESPRIT, Tunis, Tunisia*

Keywords: Social Networks, Attack, Coalition, DoS, Abuse Report, Coalition.

Abstract: In Social Network websites, the users can report the bad behaviors of other users. In order to do so, they can create a kind of escalation ticket called abuse report in which they detail the infraction made by the "bad" user and help the website moderator to decide on a penalty. Today Social Networks count billions of users, the handling of the abuse reports is no more executed manually by moderators; they currently rely on some algorithms that automatically block the "bad" users until a moderator takes care of the case. In this paper we purport to demonstrate how such algorithms are maliciously used by attackers to illegally block innocent victims. We also propose to automate such an attack to demonstrate the big damage that can be caused in current social network websites. We also took the case study of Facebook as proof of concept.

1 INTRODUCTION

Social networks (SNs) are strongly influencing the daily life of millions of citizens, companies, administrations, universities, etc. Initially, such online communities were designed to virtualize the networking activities and to facilitate social interactions between people. Unfortunately, as for any computing systems, various malicious behaviors appear aiming to corrupt the standard execution process of such a system. SNs are managing huge amount of personal and professional data, attracting cybercrime and cyber terrorism organizations to maliciously exploit such sensitive information. Recently, some political and ideological groups used SNs to attack specific SNs user profiles (Morozov, 2011) in order to isolate them from any virtual activity. This attack exploits a common vulnerability in the abuse reporting systems of the most popular SNs. Very popular SNs count hundreds of million users, it is clear that the moderation task to manage abuse reports cannot be done manually. This is why SN sites use abuse management algorithms to handle, filter and categorize the reports. Then, if need be, these reports are escalated to the moderation team to take a decision based on a human analysis. One basic and naïve algorithm consists in blocking a user profile, a page, or a group after receiving a specific number of abuse reports targeting them. The targeted element will be blocked until a human moderator studies the case. In order to perform this attack the criminal organization can rely on an important number of attackers. This is what we categorize under the umbrella of coalition attack (Srivatsa, 2005). In this paper, we propose to automate this attack in order to get rid of the necessity of having a large attacking community of a criminal organization. The goal, of our attack tool is of course not to help malicious users to perform such DoS attacks, but to explain and proof that such attack can be easily executed over thousands victim profiles with a simple mouse click that takes only a few seconds. We thus claim that it is crucial to the SNs security officers to change and upgrade their automatic abuse reporting management algorithms. As proof of concept, we tested our attacking tool in Facebook (Facebook, 2012) as the most popular SN in the world. The execution of the tool gives some interesting hints on how the abuse reporting system is working in Facebook. We analyze most of the reporting parameters in order to propose some possible countermeasures.

This paper is structured as follows: in section II we give an overview of the current state of the art related to the coalition attack exploiting social networks aspects, in section III we explain the theoretical aspect of the coalition attack, in section IV we detail our proof of concept that automates this

Trabelsi S. and Bouafif H..
Abusing Social Networks with Abuse Reports - A Coalition Attack for Social Networks.
DOI: 10.5220/0004523005000505
In *Proceedings of the 10th International Conference on Security and Cryptography* (SECRYPT-2013), pages 500-505
ISBN: 978-989-8565-73-0
Copyright © 2013 SCITEPRESS (Science and Technology Publications, Lda.)

attack in a real SN, in section V we propose a set of countermeasures to prevent the different variants of this attack.

2 RELATED WORK

Abuse reporting systems in SNs can be defined as a specific application of the traditional reputation system. To be more precise, it refers to negative feedback reputation systems (Tennenholtz, 2004) where reporting users are only solicited to declare negative behaviors of other users. As any reputation system, the abuse reporting algorithms are vulnerable to what we call coalition attacks, also called orchestrated attacks (Srivatsa, 2005), and when this attack is automated as we show it in this paper it becomes a DoS attack. Hoffman et al (Hoffman, 2007) clearly defined and distinguished these different levels of attacks in reputation systems. The bridge between generic reputation systems and social organization was initially pointed out by (Cristani, 2011), inspired by the study on logic for coalition power in game theory defined by Pauly in (Pauly, 2002) and (Pauly, 2001). (Cristani, 2011) formally proved that in social groups, a coalition of agents can behave in a malicious way in order to corrupt the normal behavior of the system. Such approach was declared as generic, but it is clearly not applicable for the specificities of SNs where several parameters are not taken into account (relationships between users, shared virtual ideology, group memberships, etc.). A spontaneous solution was proposed by (Slashdot, 2012) contributors called "Crowdsourcing the Censors: A Contest" (Von Ahn, 2003). It was proposed in order to delegate the administration and moderation task to group of SN users. With this approach the number of volunteers should be sufficient to give a response to all the abuse report requests. Attackers can also be part of these volunteers, but they assume that if the number of volunteers is huge and the selection of the user is random they will not affect the process. This subjective solution has two main limitations: the language and the cultural background of the volunteers are not systematically adapted to the content of the report. When abuse report is claiming that the targeted user account is fake. How a simple user can verify this statement?

To our knowledge, there has not been yet any scientific study on the impact of coalition attacks in SNs, nor an attempt at automating this attack to become a powerful nor damaging DoS tool for social communities.

3 COALITION ATTACK PRINCIPLE FOR SOCIAL NETWORK USERS BLOCKING

The ideologically based coalition attack represents the manual approach for a DoS rush on the abuse system of the SN. This kind of attack requires an important investment in manpower and a certain complexity in synchronizing these actors. In order to formalize such attack, we propose to represent the user interaction description of social networks in a basic model defining a subset elements related to a generic social network.

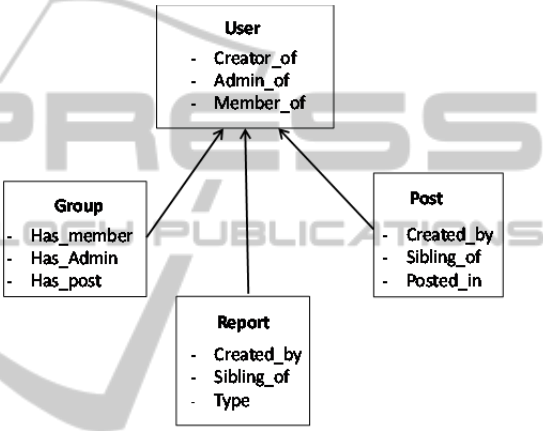

Figure 1: Basic user model for social networks.

The terminology related to this model is described below:

- User: is a virtual profile social network person member of the online community. This user can create and administrate groups, be member of several groups, publish posts and abuse reports.
- Group: is a set of users sharing a common interest. The group of users may be structured as a hierarchical community with a group creator or a group administrator.
- Post: a post is a message or a text posted by the user in the social network. This post can be sibling or referencing a user.
- Report: a report is an abuse or warning message that notifies the illegal behavior of a user or a non-ethical post sent by a user or a group of users. The report is created by a user and sibling a user or a post.

Let's assume that $\Re$ the group of N abuse reports r sibling the user s where:

$$\Re = \{r_1, r_2, r_3, ..., r_N\} \mid r.sibling\_of = s$$

We will later observe that this number N can vary in certain SNs according to the "popularity of s"

The attack becomes malicious, coordinated and ideologically motivated if the authors of $\Re$ are

sharing the same preference groups. We call these groups Φ that gather all the user authors of the reports $\Re$. Φ is the set of users that reported the abuse against *s*. The attack must be coalition is in this group all or most of the members share a secret or a motivation.

$$\Phi = \{u_1, u_2, u_3, ..., u_N\} \mid u_i = r_i.creator$$

Where $s \notin \Phi$ this is due to the fact that we consider that the targeted victim should not be member of these ideological groups as he is designed as enemy.

Starting from this model we will explain later on how to detect and prevent such a "manual" attack by performing internal checking from the SNs abuse report engines.

The detection task become more complex if the attacker user profiles are not human, generated on the fly without sharing any common interest or ideology.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

4 AUTOMATING THE COALITION ATTACK: THE FACEBOOK SOCIAL NETWORK AS A CASE STUDY

As a proof of concept for this vulnerability we decided to target the attack to the most popular SN nowadays: Facebook (Facebook, 2012). With more than 1 billion active users spread all over the world, Facebook is a very good playground to test our approach. This SN is also reputed to be one of the most secure against external attacks, and relies on a very dynamic code deployment that makes that reverse engineering process very complicated.

4.1 Basic Abuse Report Process

We describe in this section the usual process of an abuse report action sent by a normal user (see Fig2). In addition to the basic look and feel interface shown to the user we give you some information about the metadata exchanged between the Facebook server and the user's browser.

1. Once the user access Facebook's URL and display it, a new cookie is initialized and stored in the browser's cache.
2. The user authenticates himself with his credentials
3. Display of the user's page
4. A new sessions is active

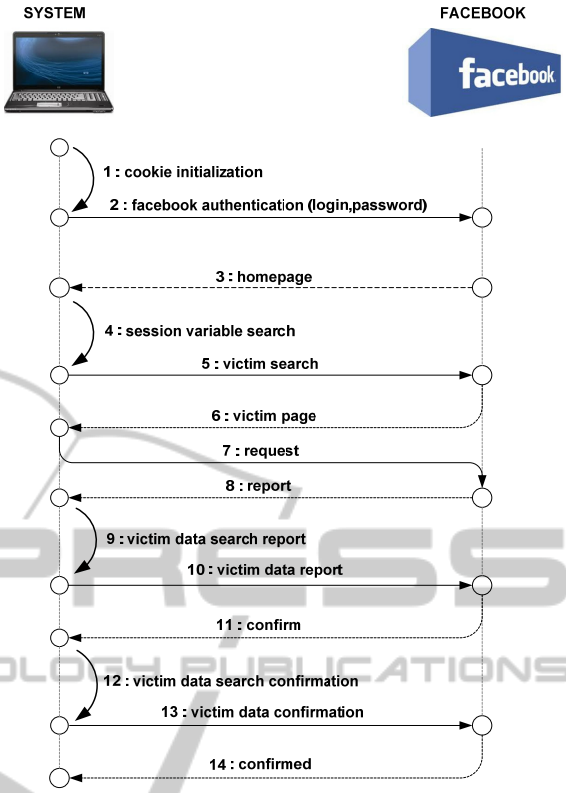

Figure 2: Abuse report process in Facebook.

5. The user can search for a non-friend target user profile or request access to a friend profile.
6. Once the targeted profile is selected the webpage depicting his information is displayed in the browser
7. The user requests an abuser report in order to notify a bad behavior of the target profile
8. A report Ajax window is then displayed in the user's browser
9. The user selects the type of infraction he wants to report
10. The report is sent to the Facebook server
11. A confirmation request is sent back to the user (in case of mistake)
12. Once the report is confirmed it becomes an active abuse report triggered on Facebook's server.

The user in Facebook has the possibility to choose several types of Abuse reports. According to our experience the type that generates the highest process time during the report treatment from Facebook administrators is the "This timeline is pretending to be someone or is fake". This long delay is due to the verification procedure; the target

account can be blocked until the owner of the profile proves his identity. This verification delay is added to the treatment delay of an abuse report.

The coalition attack consists then to synchronize with a group of users to perform the same action more or less simultaneously on a common profile target with the report type "fake account". After N reports Facebook report abuse handling algorithm will automatically block the target account until a human administrator comes and verifies the legitimacy of the reports. The target profile owner will then be contacted by e-mail and requested to provide a copy of his ID card or passport to prove his identity.

4.2 Manual Simulation of the Coalition Attack

In order to replay the attack described in the previous section, we created a dummy testing account that will play the role of victim, then we asked to our colleagues to join us in a coalition attack. For the sake of efficiency, we synchronized our attack in order to be all in all executed in a window time frame of 4 hours. The coalition was composed of 44 volunteer Facebook users. The attack was executed during 5 hours. After the last abuse report was sent, the target account was blocked. The lesson learned from this experience, is that 44 reporters is maybe lower than the minimum threshold *N*. Our attack was at the end successful, but we don't have any guarantee or proof that the targeted account was blocked by the automated abuse report system and not by a human administrator. The next step is to automate and simulate the attackers in order to prove the attack and try to identify N.

4.3 DoS Attack for the Abuser Reporting System: Automating the Coalition Attack

4.3.1 Analyzing an Abuse Report Request

In order to simulate an abuser reporting action, we have to study all the parameters exchanged between the browser and the Facebook server through HTTP. We captured and analyzed all the HTTP traffic generated during one report abuse action. We used the Zed Attack Proxy (OWASP, 2012) from OWASP that offers a powerful web application scanner monitoring, interception and modifying the different parameters of the application.

This is the list of variables that are sent to the server during an abuse report action. These variables are requested by a Javascript form. (This list may change over time according to the releases of the Facebook API):

- phase_branch: abuse treport type (fake account, impersonation of id, etc.).
- authentic_uid: impersonation of the uid.
- impersonated_user_name: fake user name.
- duplicate_id: impersonation.
- sub_lost_access: ?
- sub_fake_profile: sub option of fake account.
- rid: facebook identifier of the victim.
- cid: facebook identifier of the victim.
- hour: the hash of the reporting link.
- content_type: the content type of the request.
- are_friends: check of the relation of friendship between both profiles.
- is_ following: check of the relation of following between both profiles.
- time_flow_started: time of reporting starting
- is_tagged: check if both accounts share photos in common.
- on_profile: check of a profile relationship between both accounts
- duplicate_id: check if there is duplication of id.
- ph: ?
- phase: defines the reporting phase.
- expand_report: if yes, a second window of reporting complement will appear.
- nctr [_mod]: location of the reporting button differs in case of a timeline account.
- __ d: request limiter.
- fb_dtsg: variable of session.
- __ user: attacker Facebook identifier.
- phstamp: equal to the sending request hashing

All these variables are not officially documented by Facebook, then we proposed our definition. The parameters that we did not defined (marked by a "?") are optional, can be replayed (are not dependent from any timestamp or session ID).

4.3.2 Generating Fake Abuse Report Messages

Generating automated abuse report consists in Filling the user ID and the targeted profile ID. The rest of the variables are just replayed as received from the server (see Fig3.). Here is a non-exhaustive list of variables that we replayed during the attack: The variables *ph*, *phstamp* , *post_form_id* and *fb_dtsg* do

not change although the session changes. The variable *time_flow_started* is composed of the timestamp and a sequence number that is incremented by one every request.

The challenging part of the attack is the creation of dummy attackers profiles then the login. The creation of these accounts was quite tricky due to the

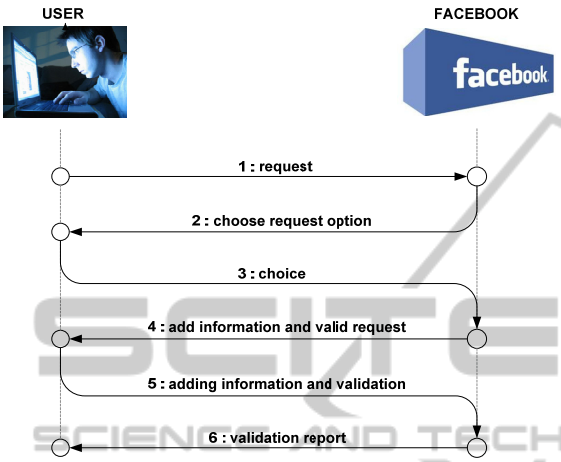

Figure 3: Automating the abuse reporting.

confirmation request that is mandatory to validate a new account. The absence of client-side challenge-response mechanisms made this task easier. The second "vulnerability" concerns the possibility to execute two actions for a new account before being asked for a confirmation. These two actions can be abuse report sending. These weaknesses are quite enigmatic, because nowadays most of the websites are preventing the automatic creation of accounts.

4.3.3 Result of the Attack

The setup phase of the attack is the creation of fake user accounts. Then, fake reports are sequentially sent until the victim account is blocked. We performed several tests on fake victim accounts, with very limited activities and a small number of virtual "friends" in the SN. All of these accounts where blocked in few seconds. The only observation that we made, is that more a user account is "popular" more *N* is large. Popular profile means with a lot of friends and activity (publications). During our tests we did not execute the attack against real users for ethical reasons. We are currently trying to create several fake accounts with a certain popularity to have a wider scope in terms of testing profiles.

5 PREVENTING USER BLOCKING ATTACKS IN SOCIAL NETWORKS

In this section we provide two main solutions to prevent such attacks in SNs.

5.1 Preventing Coalition Attacks

As defined at the beginning of the paper the manual collation attack, mainly motivated by ideological reasons, and executed by a group of SN users that agreed to block a specific user profile. The victim is not chosen randomly, but most of the time chosen according to the adversity of his ideological beliefs. Starting from this observation, an SN administrator can extract some information that can be useful to differentiate between a real abuse report flow and a coalition attack. We re-use the formal notation depicted in section.

We already defined $\Re$ as the set of reports sibling the target profile *s*. Φ is the group of users that reported the abuse against s in a certain constrained period of time (that must be defined). We propose a list of verifications to execute before blocking the targeted profile.

- G_i is the list of social groups to which belong all the users in Φ
  $G_i = \{G_1^i, G_2^i, G_3^i, ...\} = u_i.subscriber$
- *GI* is the intersection of all the groups G_i representing the common social interests of the users in Φ.
  $GI = \bigcap_i (G_i)$
- The common ideological interest can be automatically detected at this stage if most of the reporters share a social interest or belongs to the same interest group. If the reporters mainly belong to the same groups. δ is a threshold value defined by the SN administrator that identifies the users sharing the same interests. The targeted profile *s* must not belong to these groups.
  $GI.count(\Phi) > \delta$
- The targeted profile *s* must not belong to one of these groups.
- If in these interest groups a link to *s* profile is detected this can be the proof that the victim was added in a kind of black list of users the must be blocked. This check point is optional.

The solution proposed here tends to identify the common interest of the attacker with some analytics queries. Making these verifications before blocking the target profile can be beneficial to the victims of the coalition attacks. Of course these checking should not prevent any human intervention of the SN moderators to evaluate the credibility of a set of abuse reports. We also propose a reputation and penalties system for the attackers if the SN moderator detects the coalition attack via our solution. A kind of caution message can be sent to the suspected attacker to warn him against a fake abuse report action.

5.2 Preventing Automated DoS Attack

It is clear that the execution of the automated DoS attack is more powerful and damaging than the coalition attack. Few seconds are sufficient to block any profile. The countermeasure is less complex than the one proposed for the coalition attack. The traditional client-side security challenge response tests (Mirkovic, 2004) are to our opinion the most appropriate solution. These solutions must be applied during the creation of new SN accounts, then during the abuse response sending. The challenges will limit the creation of fake accounts, and the generation abuse reports without a human intervention.

6 CONCLUSIONS

In this paper we have formally identified a serious vulnerability in the abuse reporting systems that are currently deployed in most of the SN websites. We first observed the problem in the real world where ideological groups of users in different SNs are permanently setting up coalition attacks based on a particular misuse of the abuse reporting systems in order to block other innocent users that are judged as ideological enemies. We provided a technical analysis of this attack then we proposed to automate it in order to exploit this vulnerability through a DoS attack. We developed a proof of concept exploiting this vulnerability in the SN website Facebook that is also considered as one of the most secure. Although incomplete, the first results obtained clearly demonstrate the damages that can be caused by such DoS tools, especially if we upgrade the attack from a DoS to a DDoS where (executing the attack simultaneously). We propose two different approaches to prevent against such attacks and specially the coalition attack. The study is still at its initial phase, we are not yet able to clearly define the variable *N* representing the number of abuse reports that will automatically block a user profile. More advanced tests are currently executed to explore all the dimensions of this vulnerability.

ACKNOWLEDGEMENTS

The research leading to these results has received funding from the European Community's Seventh Framework Program in the context of PPP Fi-Ware project and the EIT – KIC Trust in the Cloud EU Project.

REFERENCES

Morozov, E., 2011. "*The Net Delusion: The Dark Side of Internet Freedom*" (New York: Public Affairs, 2011)

Srivatsa, M., Xiong, L., Liu, L., 2005. "TrustGuard: Countering Vulnerabilities in Reputation Management for Decentralized Overlay Networks", *Proceeding of the 14th international conference on World Wide Web* (WWW'05), New York, USA

Hoffman, K., Zage, D., Nita-Rotaru, C., 2007. "A Survey of attacks on Reputation Systems", *Computer Science Technical Report* – Number 07-013 – Perdue University (2007)

Tennenholtz, M., 2004 "Reputation Systems: An Axiomatic Approach", UAI *'04 Proceedings of the 20th conference on Uncertainty in artificial intelligence - Pages 544-551 - AUAI Press Arlington,* Virginia, United States 2004.

Cristani, M., Karafili, E. and Viganò, L., 2011. "Blocking Underhand Attacks by Hidden Coalitions", *3rd International Conference on Agents and Artificial Intelligence*, Rome, Italy, 28-30, ICAART 2011.

Pauly, M., 2002 "A modal logic for coalition power in games". *Journal of Logic and Computation*, 12(1):149–166. (2002)

M. Pauly, "*Logic for social software"*. PhD thesis, Institute for Logic Language and Computation, University of Amsterdam. (2001)

Facebook, https://www.facebook.com/

Von Ahn, L., Blum, M., Hopper, N. and Langford, J., 2003. "CAPTCHA: Using Hard AI Problems for Security". *In Proceedings of Eurocrypt 2003, May 4-8,* 2003, Warsaw, Poland.

Mirkovic, J. and Reiher, P., 2004. "A taxonomy of DDoS attack and DDoS defense mechanisms". *ACM SIGCOMM 2004*, Comput. Commun.

OWASP Zed Attack Proxy Project

Slashdot, http://tech.slashdot.org/story/11/04/15/1545213/Crowdsourcing-the-Censors-A-Contest