Key Challenges in Defending Against Malicious Socialbots

Yazan Boshmaf, Ildar Muslukhov, Konstantin Beznosov, Matei Ripeanu
University of British Columbia, Vancouver, Canada

Abstract

The ease with which we adopt online personas and relationships has created a soft spot that cyber criminals are willing to exploit. Advances in artificial intelligence make it feasible to design bots that sense, think and act cooperatively in social settings just like human beings. In the wrong hands, these bots can be used to infiltrate online communities, build up trust over time and then send personalized messages to elicit information, sway opinions and call to action. In this position paper, we observe that defending against such malicious bots raises a set of unique challenges that relate to web automation, online-offline identity binding and usable security.

1 Introduction

Our personal and professional lives have gone digital: we live, work and play in cyberspace. We use the Web every day to talk, email, text and socialize with family, friends and colleagues. Yet, this new social Web, which is predominated by Online Social Networks (OSNs), is not exclusive to human beings.

A new breed of bots called *socialbots* are on the rise, and they can be used to influence OSN users. A socialbot is an automation software that controls an OSN account and has the ability to execute basic social activities such as posting a message or sending a friendship request. What is special about a socialbot is that it is designed to be stealthy, that is, it is able to pass itself off as a human being. This is achieved by either simply mimicking the actions of a real OSN user or by simulating such a user using artificial intelligence, just as in social robotics. Project Realboy [4], for example, aims to design believable Twitter bots that imitate real users.

The initial objective of operating socialbots was non-adversarial. The Web Ecology Project [19], for example, envisions the design of socialbots that have positive impact on online communities, where bots are used to advocate awareness and cooperation among human users on civic or humanitarian issues. Soon after, this objective got extended towards *social architecture* [14]: the technology where socialbots are used to interact with, promote and provoke online communities towards desirable behaviors, including large-scale restructuring of social graphs. Unfortunately, in the wrong hands, these socialbots can be used for adversarial objectives as well.

We recently showed that it is feasible to run a large-scale *infiltration* campaign in a target OSN with a high success rate [3], where an adversary orchestrates a network of socialbots in order to connect to a large number of users. A successful infiltration campaign has the following serious security implications:

First, the social structure of the target OSN can be compromised and polluted with a large number of non-genuine social relationships. This means that it is unsafe to treat the infiltrated OSN as a strong trust network. Therefore, third-party applications and websites have to perform appropriate “clean up” to identify and remove most of the bogus users along with their fake relationships, all before integrating or using such an OSN [24].

Second, in addition to online surveillance, the adversary can breach the privacy of the infiltrated users by harvesting large amounts of private data, including personally identifiable information such as email addresses, phone numbers and birth dates, all of which have considerable monetary value in the Internet’s underground markets. Moreover, this data can be used to run follow up and highly personalized email spam or phishing campaigns, which usually have a high success rate [8].

Third and last, the adversary can exploit the infiltrated OSN to spread misinformation [15], distribute social malware and other malicious content [22], or even influence algorithmic trading that uses opinions extracted from OSNs to predict the stock market [2].

These alarming implications indicate that protecting OSNs from malicious socialbots is important not only to users, but also to OSN operators and online social media businesses. Still, the objective of this position paper is not to argue the importance of social network security, as existing research already makes this case [16]. Instead, we observe that defending against malicious socialbots involves solving challenges related to web automation, online-offline identity binding and usable security.

We recast the definition of a socialbot as an automated social engineering tool in Section 2. Treated that way, we briefly outline how such bots exploit inherent vulnerabilities found in today’s OSNs that allow an adversary to automate their operation. After that in Section 3, we argue that existing OSN security defenses are not effective enough at detecting socialbots and do not eliminate the factors that make operating such bots feasible in the first place. This provides the necessary background for the key challenges we present in Section 4.

2 Automated Social Engineering

Huber et al. [7] presented one of the first frameworks for automated social engineering in OSNs, where a new breed of bots can be used to automate traditional social engineering attacks for many adversarial objectives. Given this context, a malicious socialbot can be thought of as an automated social engineering tool that allows an adversary to infiltrate online communities like a virtual con man, build up trust over time and then exploit it to elicit information, sway opinions and call to action.

In fact, automation has a strong economic rationale behind it. Herley [6] shows that for an online attack to be scalable, it ought to be automated without manual per-user adjustments. Otherwise, there are no economic incentives for a rational adversary to scale the attack, which is undesirable from an adversarial standpoint.

Today's OSNs suffer from inherent vulnerabilities that can be exploited to allow such automation [3]. In particular, most OSNs employ ineffective CAPTCHAs, allow multiple accounts to be created by the same user, hide the social graph but permit any user to crawl it and provide social APIs and platforms that are relatively easy to exploit or reverse engineer. Collectively, along with poorly designed end-user privacy controls [9], these vulnerabilities represent the *enabling factors* that make operating socialbots feasible in the first place.

3 Social Network Security

A number of recently proposed techniques aim to automatically identify bots in OSNs based on their abnormal behavior [17, 23]. Facebook Immune System [16], for example, performs real-time checks and classification on every read and write action on Facebook's database, all for the purpose of protecting its users and the social graph from malicious activities. It is thus not surprising that such an adversarial learning system is rather effective at identifying and blocking *spambots*: malicious bots that use fake or hijacked OSN accounts to vastly distribute unsolicited messages to non-consenting users.

Socialbots, however, are much more deceptive than spambots as they are designed to appear "normal" [3]. Armed with today's artificial intelligence advancements, it is feasible to orchestrate a network of socialbots that sense, think and act cooperatively just like human beings. An adversary, for example, can employ adversarial classifier reverse engineering techniques [10] in order to learn sufficient information about the security defenses deployed by the target OSN, and then construct an adversarial strategy that minimizes the chance of a socialbot being detected, sometimes down to zero.

Graph theoretic techniques, as an alternative to adversarial learning systems, are less effective and more expensive at identifying socialbots, as one would typically "look for a needle in a haystack." Community detection algorithms, for example, are deemed to fail as there will be far more fake relationships than socialbots [24]. The intuition behind this is that each socialbot is expected to gradually, but independently, integrate into the online community it targets, resembling the scenario when a new user joins an OSN and starts befriending others.

To that end, the problem of detecting malicious socialbots in OSNs is similar to filtering email spam: it is an arms race and will keep both the defenders and the attackers busy, depending on their available resources. Fortunately, unlike email spam, malicious socialbots can be effectively blocked by eliminating the enabling factors that make them feasible. Doing so, however, involves solving the following socio-technical challenges.

4 Challenges

We now present the key challenges in defending against malicious socialbots. Our objective is not to reduce the severity of the problem, as existing security defenses achieve this [16]. Instead, we aim to eliminate the factors that cause the problem in the first place, that is, to fix one or more of the vulnerabilities outlined in Section 2.

4.1 Web Automation

To simulate a user browsing an OSN, the adversary can use *web automation*, which includes methods for solving CAPTCHAs, creating and populating multiple OSN accounts, crawling the social graph and executing online social activities. Preventing this automation, however, requires solving at least one of the following challenges.

Challenge 1. *Design a reverse Turing test that is usable and effective even against "illegitimate" human solvers.*

A *reverse Turing test*, such as CAPTCHA, is a test that is administered by a machine and is designed to tell humans and machines apart [21]. A *perfect* test presents a problem that is easy enough for all humans to solve, but is still impossible for a machine or an automation software to pass. Unfortunately, even a perfect test is ineffective if humans are exploited to solve the test in an *illegitimate setting*: the situation where human users are employed or tricked into solving reverse Turing tests that are not addressed to them. Under this setting, we refer to such a human user as *illegitimate*.

Eliminating the economic incentives for underground businesses that employ illegitimate human solvers is a first step towards tackling this challenge [13], but it does not solve it as legitimate users can still be tricked and situated into illegitimate settings, which is the case for the Koobface botnet [1]. This demands the design of new reverse Turing tests that are resilient to even those illegitimate users, which we believe is difficult to achieve.

Fast-response CAPTCHAs, for example, require the test to be solved in a relatively shorter time, as opposed to typical implementations. This makes it more difficult for automation scripts to pass the test, as they require extra time to relay the test, solve it and respond back. Fast-response CAPTCHAs, however, are expected to put more pressure on legitimate users who require easy and fast access to online services, and could potentially repel them away from using them.

Challenge 2. *Effectively limit large-scale Sybil crawls of OSNs without restricting users' social experience.*

A *large-scale crawl* is a malicious activity where an adversary manages to crawl large portions of a target OSN, including both the social graph and all accessible users' profile information. Today, large-scale crawls are mitigated by employing a network-wide audit service, which limits the number of profiles a user can view per account or IP address in a given period of time [16]. This, however, can be circumvented by using a set of accounts, each called a *Sybil*, and then performing *Sybil crawling* on a large scale, typically using a botnet with multiple IP addresses [1].

To overcome this drawback, one can use the knowledge about the social graph to effectively limit Sybil crawls. Genie [12], for example, is a system that models the trust between users in an OSN as a *credit network*, where a user can view the profile of another user only if the path between them in the social graph has enough credits to satisfy the operation. If an adversary who controls many Sybil accounts attempts to crawl the OSN on a large scale, then Genie guarantees that the adversary will exhaust all the credits on the paths connecting the Sybil accounts to the rest of the network, thus limiting large-scale Sybil crawls. This approach, however, is based on the assumption that it would be hard for an adversary to establish an arbitrarily large number of social relationships with other users, which we showed to be an unsafe assumption, especially in OSNs as Facebook [3].

Challenge 3. *Detect abusive and automated usage of OSN platforms and social APIs across the Internet.*

In concept, *malicious automation* represents the situation where an adversary scripts her way of consuming system's resources in order to cause damage or harm to the system itself or its users. *Abusive automation*, on the other hand, is less severe where the adversary exploits the offered service in violation of the declared Terms of Service. From the OSN operator standpoint, all HTTP requests come from either a browser or through the social API, which is intentionally provided to support automation. Requests that are not associated with a browsing session, that is, those that do not append the required session cookies, can be easily detected and dealt with. With web automation, however, an adversary can simulate an OSN user and make all requests look as if they originate from a browser. Moreover, the patterns at which these requests are made can be engineered in such a way that makes them fall under the normal traffic category [3]. In order to uncover adversarial campaigns, it is important to reliably identify whether such requests come from a human or a bot, along with means to distinguish patterns of abusive activities, even if the adversary has a knowledge of the used classification techniques.

Looking for regularities in the times at which requests are made, for example, can be used to detect automation in OSNs [25]. This, however, can be easily circumvented by simply mimicking the times and irregularities at which a human user makes such requests.

4.2 Identity Binding

Most of the challenges we presented so far are difficult due to the capability of the adversary to mount the *Sybil attack* [5]: an attack where an adversary controls multiple online identities and joins a targeted system under these identities in order to subvert a particular service. This leads us to the following challenge:

Challenge 4. *Guarantee an anonymous, yet credible, online-offline identity binding in open-access systems.*

A system is called *open-access* if it allows any user to join the system by providing an identity that is issued by the system itself or by other identity providers [24]. Douceur [5] shows that without a centralized trusted party that certifies online identities, Sybil attacks are always possible except under extreme and unrealistic assumptions of resource parity and coordination among participating entities. Thus, limiting the number of Sybil accounts by forcing a clean mapping between online and offline identities is widely recognized as a hard problem, especially given the scalability requirements of today's online, open-access software systems.

Arguably, one way to tackle this challenge is to rely on governments for online identity management just as in offline settings. The *open government* initiative [20], for example, enables U.S. citizens to easily and safely engage with U.S. government websites using open identity technologies such as OpenID. This, however, requires creating *open trust frameworks* [20] that enable these websites to accept identity credentials from third-party identity providers, a task that involves solving challenging issues related to identity anonymity, scalability, security, technology incentives and adoption [11, 18].

4.3 Usable Security

As part of computer security, *usable security* aims to provide the users with security controls they can understand and privacy they can control. In OSNs such as Facebook, there appear to be a growing gap between what the user expects from a privacy control and what this control does in reality [9]. Even if the most sophisticated OSN security defense is in place, an OSN is still vulnerable to many threats, such as social phishing [8], in case its users find it puzzling to make basic online security or privacy decisions. This gives us strong motives to study the human aspect of the OSN security chain, which is by itself a challenge.

Challenge 5. *Develop usable OSN security and privacy controls that help users make more informed decisions.*

Designing security controls that better communicate the risks of befriending a stranger, for example, might be practically effective against automated social engineering. This, however, requires eliciting and analyzing the befriending behavior of users, including the factors that influence their befriending decisions, in order to inform a user-centered design for such controls.

5 Conclusion

From a traditional computer security perspective, the concept of socialbots is both interesting and disturbing: the threat is no longer from a human controlling or monitoring a computer, but from exactly the opposite. As with other online attacks, defending against malicious socialbots is an arms race where the objective of the defender is to limit any potential harm or damage, that is, to extend the time at which the system enjoys its safe state. In this paper, we observed that in order to effectively defend against such bots, one has to fix a set of inherent vulnerabilities found in today's OSNs, which collectively represent the enabling factors causing the problem. This, however, boils down to solving a number of socio-technical challenges that relate to web automation, online-offline identity binding and usable security.

References

[1] J. Baltazar, J. Costoya, and R. Flores. The real face of Koobface: The largest web 2.0 botnet explained. *Trend Micro Research*, July 2009.

[2] J. Bollen, H. Mao, and X. Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1 – 8, 2011.

[3] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu. The socialbot network: when bots socialize for fame and money. In *Proceedings of the 27th Annual Computer Security Applications Conference*, ACSAC '11, pages 93–102, New York, NY, USA, 2011. ACM.

[4] Z. Coburn and G. Marra. Realboy: Believable twitter bots. `http://ca.olin.edu/2008/realboy`, April 2011.

[5] J. R. Douceur. The sybil attack. In *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, pages 251–260, London, UK, 2002. Springer-Verlag.

[6] C. Herley. The plight of the targeted attacker in a world of scale. In *The 9th Workshop on the Economics of Information Security (WEIS 2010)*, 2010.

[7] M. Huber, S. Kowalski, M. Nohlberg, and S. Tjoa. Towards automating social engineering using social networking sites. *Computational Science and Engineering, IEEE International Conference on*, 3:117–124, 2009.

[8] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer. Social phishing. *Commun. ACM*, 50(10):94–100, 2007.

[9] Y. Liu, K. P. Gummadi, B. Krishnamurthy, and A. Mislove. Analyzing facebook privacy settings: user expectations vs. reality. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, IMC '11, pages 61–70, New York, NY, USA, 2011. ACM.

[10] D. Lowd and C. Meek. Adversarial learning. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, KDD '05, pages 641–647, New York, NY, USA, 2005. ACM.

[11] E. Maler and D. Reed. The venn of identity: Options and issues in federated identity management. *IEEE Security and Privacy*, 6:16–23, 2008.

[12] M. Mondal, B. Viswanath, A. Clement, P. Druschel, K. P. Gummadi, A. Mislove, and A. Post. Limiting large-scale crawls of social networking sites. In *Proceedings of the ACM SIGCOMM 2011 conference on SIGCOMM*, SIGCOMM '11, pages 398–399, New York, NY, USA, 2011. ACM.

[13] M. Motoyama, K. Levchenko, C. Kanich, D. McCoy, G. M. Voelker, and S. Savage. Re: Captchas: understanding captcha-solving services in an economic context. In *Proceedings of the 19th USENIX Conference on Security*, USENIX Security'10, pages 28–28, Berkeley, CA, USA, 2010. USENIX Association.

[14] M. Nanis, I. Pearce, and T. Hwang. Pacific social architecting corporation: Field test report. `http://pacsocial.com/`, November 2011.

[15] J. Ratkiewicz, M. Conover, M. Meiss, B. Gonçalves, S. Patil, A. Flammini, and F. Menczer. Truthy: mapping the spread of astroturf in microblog streams. In *Proceedings of the 20th international conference companion on World wide web*, WWW '11, pages 249–252, New York, NY, USA, 2011. ACM.

[16] T. Stein, E. Chen, and K. Mangla. Facebook immune system. In *Proceedings of the 4th Workshop on Social Network Systems*, SNS '11, pages 8:1–8:8, New York, NY, USA, 2011. ACM.

[17] G. Stringhini, C. Kruegel, and G. Vigna. Detecting spammers on social networks. In *Proceedings of the 26th Annual Computer Security Applications Conference*, ACSAC '10, pages 1–9, New York, NY, USA, 2010. ACM.

[18] S.-T. Sun, Y. Boshmaf, K. Hawkey, and K. Beznosov. A Billion Keys, but Few Locks: The Crisis of Web Single Sign-On. In *Proceedings of the New Security Paradigms Workshop (NSPW)*, pages 61–72, September 20–22 2010.

[19] The Web Ecology Project. The 2011 socialbots competition. `http://www.webecologyproject.org/category/competition/`, January 2011.

[20] D. Thibeau. Open trust frameworks for open government: Enabling citizen involvement through open identity technologies. `http://openid.net/government/`, August 2011.

[21] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford. Captcha: Using hard AI problems for security. In *EUROCRYPT*, pages 294–311, 2003.

[22] G. Yan, G. Chen, S. Eidenbenz, and N. Li. Malware propagation in online social networks: nature, dynamics, and defense implications. In *Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security*, ASIACCS '11, pages 196–206, New York, NY, USA, 2011. ACM.

[23] Z. Yang, C. Wilson, X. Wang, T. Gao, B. Y. Zhao, and Y. Dai. Uncovering social network sybils in the wild. In *IMC*, Berlin, Germany, November 2011.

[24] H. Yu. Sybil defenses via social networks: a tutorial and survey. *SIGACT News*, 42:80–101, October 2011.

[25] C. M. Zhang and V. Paxson. Detecting and analyzing automated activity on twitter. In *Proceedings of the 12th international conference on Passive and active measurement*, PAM'11, pages 102–111, Berlin, Heidelberg, 2011. Springer-Verlag.