Journal Homepage: www.ijarpr.com ISSN: 3049-0103 (Online)

International Journal of Advance Research Publication and Reviews

Vol 03, Issue 9, pp 591-593, September, 2026

Identification and Classification of Malicious Third-Party Applications on Online Social Networks

[1]M. Sreedhar Reddy, [2]Haricharan, [3]Mahesh Kumar, [4]B Vaishnavi

[1,2,3,4] Department of ECE
Narsimha Reddy Engineering College

ABSTRACT:

Online Social Networks (OSNs) support a large ecosystem of third-party applications that improve user engagement but can also introduce security and privacy risks. Malicious applications may abuse granted permissions to access private information, perform clickjacking, distribute spam, and propagate additional malicious applications. This paper presents FRAppE (Facebook's Rigorous Application Evaluator), together with its lightweight FRAppE Lite variant, for identifying malicious third-party applications. The approach profiles applications using on-demand features and aggregate cross-user behavioral features and applies Naïve Bayes and Support Vector Machine (SVM) classification. The source study reports evaluation on 111,000 applications associated with more than 91 million wall posts, with FRAppE Lite reporting 99.0% accuracy and Full FRAppE reporting 99.5% accuracy.

Keywords: Online Social Networks; Malicious Application Detection; FRAppE; Naïve Bayes; Support Vector Machine; OAuth 2.0; Application Profiling; App-Nets; Social Network Security.

1. INTRODUCTION

Online Social Networks (OSNs) provide platforms where third-party developers can create integrated applications such as games, utility software, and social tools. These applications can request permissions that allow access to profile information and posting capabilities. The source paper notes that malicious developers can exploit this model to harvest private user data, execute clickjacking campaigns, and automatically publish spam.

A key limitation of several earlier defenses is that they concentrate on individual malicious posts or standalone URLs rather than identifying the application responsible for generating the malicious activity. FRAppE addresses this gap by evaluating applications directly and combining application-level indicators for proactive detection.

2. BACKGROUND AND RELATED WORK

When a user installs an OSN application, the platform can issue an OAuth 2.0 token that grants the application server specific access rights. The source paper describes a malicious application lifecycle consisting of luring users with deceptive offers, redirecting them to phishing surveys or external sites, extracting access tokens and personal metadata, automating spam posts, and cross-promoting secondary malicious applications.

The source material contrasts FRAppE with post-level approaches such as MyPageKeeper. Instead of examining only a single URL or post, FRAppE aggregates application-level signals, including permissions, application name, URL, posting behavior, and application-network relationships.

3. PROPOSED SYSTEM ARCHITECTURE AND METHODOLOGY

The proposed framework operates in two modes. FRAppE Lite evaluates instantaneous on-demand application features, while Full FRAppE combines on-demand features with cross-user aggregated behavioral metrics. The extracted feature vector is supplied to a classification engine using probabilistic Naïve Bayes and Support Vector Machines (SVM). The classifier assigns the application to a benign or malicious class.

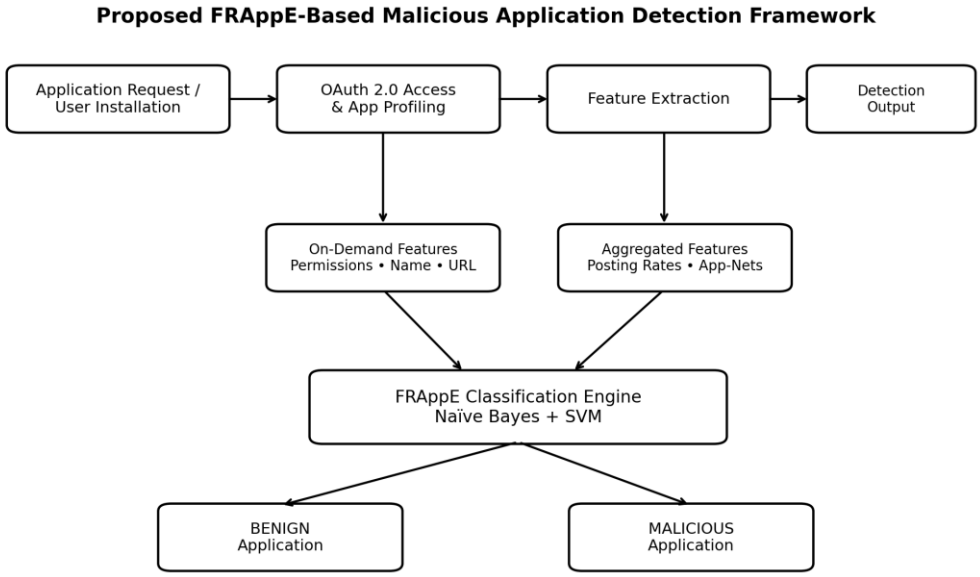

Figure 1. Block diagram of the proposed FRAppE-based malicious application detection framework.

3.1 On-Demand Features

On-demand features are available when an application is evaluated. The source paper identifies permissions, application name, and URL as examples of these indicators. These features can be used by FRAppE Lite for a lightweight decision.

3.2 Aggregated Features

Aggregated features capture behavior across users and application relationships. Examples in the source paper include posting rates and App-Net relationships, where malicious applications may form cross-promotional networks.

3.3 Classification

For an application feature vector X and class Ck, the source paper gives the Naïve Bayes decision formulation as $P(Ck \mid X) = [P(Ck) \times \prod P(xi \mid Ck)] / P(X)$. The class with the highest posterior probability is selected. SVM is also used as part of the classification framework.

4. ATTACK FLOW OF MALICIOUS APPLICATIONS

1. Lure the user through clickbait or fake rewards.
2. Redirect the user to a phishing survey or external site.
3. Extract access tokens and personal metadata.
4. Automate unauthorized viral spam posts to friends.
5. Cross-promote secondary malicious applications through App-Nets.

5. EXPERIMENTAL EVALUATION

The source study evaluates FRAppE using a dataset covering 111,000 applications associated with more than 91 million wall posts. The reported observations include dense cross-promotional networks among malicious applications, repeated application titles across different App IDs, and permission patterns that can be used as risk signals.

Framework	Accuracy	False Positive	False Negative
FRAppE Lite	99.0%	0.1%	4.4%
Full FRAppE	99.5%	0.0%	4.1%

Table 1. Detection results reported in the source paper.

6. ADVANTAGES

- Application-level detection instead of relying only on individual malicious posts or URLs.
- Combination of instantaneous and cross-user behavioral features.
- Use of Naïve Bayes and SVM classification for automated identification.
- Ability to identify App-Net relationships and other behavioral indicators.

7. LIMITATIONS AND FUTURE SCOPE

The source paper focuses on social-network applications and specifically presents Facebook-oriented FRAppE features.
Future work proposed in the source includes extending detection to cross-platform threat campaigns spanning ecosystems such as X/Twitter and LinkedIn and using deep learning techniques.

8. CONCLUSION

This paper presents a structured approach for detecting malicious third-party applications on online social networks. FRAppE combines application profiling, on-demand indicators, aggregate behavioral features, and machine-learning classification. According to the source evaluation, the framework achieved high reported classification accuracy on a large application and wall-post dataset. The block diagram summarizes the complete flow from application request and OAuth access through feature extraction, classification, and benign/malicious output.

REFERENCES

1. Facebook Open Graph API Documentation, 2016.
2. MyPageKeeper Security Platform, IEEE TPDS, 2012.
3. P. Chia et al., “Is this app safe? A large scale study on application permissions and risk signals,” Proc. WWW, 2012.
4. H. Gao et al., “Detecting and characterizing social spam campaigns,” Proc. ACM IMC, 2010.
5. M. S. Rahman et al., “Efficient and Scalable Socware Detection in OSNs,” USENIX Security, 2012.