ABSTRAK

PERBANDINGAN KINERJA ALGORITMA *NAIVE BAYES CLASSIFICATION* DAN *RANDOM FOREST CLASSIFICATION* DALAM MENDETEKSI *WEB SHELL*

Oleh

Fadli Maulana Muhammad

16/398503/PA/17464

Web shell merupakan salah satu kode program yang digunakan oleh *hacker* untuk melakukan eksploitasi pada laman web yang ditulis menggunakan bahasa pemrograman tertentu, contohnya menggunakan bahasa pemrograman *PHP*. Isi dari *Web Shell* tersebut dinamis tergantung dari pembuatnya, sehingga tiap masing-masing *Web Shell* merupakan sebuah script yang unik. Agar *Web Shell* mudah untuk diidentifikasi, perlu dikalukan konversi terlebih dahulu menjadi bentuk bahasa tingkat rendah atau *opcode* agar memiliki standar yang sama.

Beberapa algoritma yang dapat melakukan identifikasi *Web Shell* adalah *Naïve Bayes* dan *Random Forest*. Algoritma *Naïve Bayes* bekerja dengan menggunakan probabilitas dengan mempertimbangkan semua fitur secara independen satu sama lain untuk melakukan klasifikasi. Hal yang berbeda dilakukan pada algoritma *Random Forest* karena menggabungkan banyak *Decision Tree,* salah satu tujuannya adalah untuk mengurangi *overfitting.* Algoritma *Decision Tree* sendiri dapat melihat tingkat impuritas dari sebuah fitur dengan harapan dapat melihat fitur yang penting, tidak seperti pada *Naïve Bayes* yang melihat fitur secara independen. Namun, kedua algoritma tersebut baik *Naïve Bayes* maupun *Random Forest* belum diketahui algoritma manakah yang memiliki kinerja terbaik dalam mendeteksi *Web Shell*, sehingga penelitian ini akan membandingkan kinerja dari kedua algoritma tersebut.

Kedua algoritma baik *Naïve Bayes* maupun *Random Forest* memiliki performa deteksi yang baik yaitu di atas 90%. Sebagai perbandingan, performa deteksi *Random Forest* lebih baik dibandingkan dengan *Naïve Bayes* karena memiliki skor akurasi, presisi, recall, dan f1 yang lebih tinggi. Namun, *Naïve Bayes* lebih baik dibandingkan dengan *Random Forest* dalam hal waktu eksekusi karena memiliki waktu eksekusi yang lebih kecil. Dari hasil penelitian juga didapatkan bahwa *Random Forest* lebih sensitif dibandingkan *Naïve Bayes* karena memiliki selisih skor *recall* yang cukup besar yaitu 6.46% dibandingkan selisih skor presisi yang hanya sebesar 1.12%.

Kata kunci : Indentifikasi *Web Shell*, *PHP opcode, Naïve Bayes Classification, Random Forest Classification*.

ABSTRACT

PERFORMANCE COMPARISON OF *NAIVE BAYES CLASSIFICATION* AND *RANDOM FOREST CLASSIFICATION* ALGORITHMS IN DETECTING *WEB SHELLS*

By

Fadli Maulana Muhammad

16/398503/PA/17464

Web shell is one of the program codes used by hackers to exploit web pages written using a particular programming language, for example using the PHP programming language. The content of the Web Shell is dynamic depending on the author, so each Web Shell is a unique script. In order for the Web Shell to be easily identified, it is necessary to convert it first into a low-level language form or opcode so that it has the same standard.

Some algorithms that can identify Web Shell are Naïve Bayes and Random Forest. The Naïve Bayes algorithm works by using probabilities by considering all features independently of each other to perform classification. Different things are done in the Random Forest algorithm because it combines many Decision Tree, one of the goals is to reduce overfitting. The Decision Tree algorithm itself can see the impurity level of a feature in the hope of seeing important features, unlike Naïve Bayes which sees features independently. However, both Naïve Bayes and Random Forest algorithms are not yet known which algorithm has the best performance in detecting Web Shell, so this research will compare the performance of the two algorithms.

Both Naïve Bayes and Random Forest algorithms have good detection performance which is above 90%. In comparison, Random Forest detection performance is better than Naïve Bayes because it has higher accuracy, precision, recall, and f1 scores. However, Naïve Bayes is better than Random Forest in terms of execution time because it has a smaller execution time. The results also show that Random Forest is more sensitive than Naïve Bayes because it has a fairly large recall score difference of 6.46% compared to the precision score difference of only 1.12%.

Keywords : *Web Shell* Identification, *PHP opcode, Naïve Bayes Classification, Random Forest Classification*.