ARS 2018 The 4th Annual Research Seminar
Prosiding Annual Research Seminar 2018
Computer Science and ICT
ISBN : 978-979-587-813-1
Vol.4 No.1

Identifikasi *Malicious Web* Menggunakan Metode *Random Forest*

Virani Putri Perdana[1]
Fakultas Ilmu Komputer
Universitas Sriwijaya
Palembang,Indonesia
[1]viraniputri2@gmail.com

Nanda Harsana Octavya[2]
Fakultas Ilmu Komputer
Universitas Sriwijaya
Palembang,Indonesia
[2]Nandaharsana66@gmail.com

***Abstrak* — Malicious web adalah sebuah situs jahat yang dapat mengganggu operasi komputer, penjahat internet mengelabui pengguna dengan menyusupi situs web tertentu, terkadang *malicious web* sering terlihat seperti *benign web* sehingga pengguna sulit membedakan keduanya. Pada penelitian kali ini kami akan mengidentifikasi *Malicious Web* menggunakan metode *Random Forest.* Penelitian ini menggunakan 7 variabel prediksi dan 1 variabel respon . Dari hasil penelitian di dapatkan akurasinya sebesar 94% .**

Kata kunci — Identifikasi, Random Forest, Malicious Web (Web bebahaya)

I. PENDAHULUAN

Dunia teknologi semakin hari semakin berkembang tak terkecuali dunia internet. Banyak orang mempermudah suatu kegiatan menggunakan pelayanan *online* sehingga telah menarik penjahat untuk mencoba menyalahgunakan internet dan penggunanya untuk menghasilkan keuntungan ilegal[1]. Orang yang menyalahgunakan internet untuk keperluan pribadi telah menggunakan web sebagai perantara untuk menyampaikan serangan jahat seperti *phishing*, *spam*, dan infeksi *malware.* Misalnya, *phishing* biasanya melibatkan pengiriman email dari sumber tepercaya untuk mengelabui orang agar mengeklik URL yang terdapat dalam email yang tertaut ke halaman web palsu[2]. Sehingga pengguna internet tak menyadari telah masuk perangkap penjahat *malicious web.*

Beragam cara yang dilakukan penjahat *malicious web* untuk mengelabui korbannya misalnya saja serangan *malicious web* dapat dilakukan dengan menyusupi situs web tertentu yang menjadi target pengguna basis yang sering dikunjungi serta memasukan skrip berbahaya ke web konten melalui objek yang disematkan sehingga banyak pengguna web yang tertipu [3] .

Sadar akan hal ini, para peneliti keamanan telah mengembangkan berbagai cara untuk melindungi pengguna web dari pilihan mereka yang kurang informasi. Salah satunya dengan mendeteksi *malicious* URL dengan menggunakan algoritma *online.* Algoritma ini memiliki keunggulan yaitu memiliki sistem klasifikasi yang memproses umpan langsung dari URL berlabel dan mengumpulkan fitur untuk URL ini secara *real time.* Penelitian tersebut memiliki akurasi hingga 99% dan tingkat kesalahan dalam mengindentifikasi *malicious web* yaitu sebesar 1% [4].

Pada penelitian yang lain menggunakan metode *machine learning*, akurasi yang dihasilkan dari pendeteksian *malicious web* sebesar 95-99% dan tingkat kesalahan dalam mengindentifikasi *malicious web* yaitu sebesar 5%-1%[5].

Banyak teknik klasifikasi yang berkembang, *Random Forest* menjadi salah satu alternatif teknik klasifikasi dan dapat meningkatkan akurasi. Pada penelitian ini kami akan mengindentifikasi *malicious web* menggunakan metode *Random Forest* dengan dataset yang di dapat dari kaggle yang datasetnya bernama *malicious and benign websites* [6] dengan cara mengklasifikasi *malicious web* dan *benign web* tanpa memeriksa konten sebenarnya.

II. METODOLOGI

A. Dataset

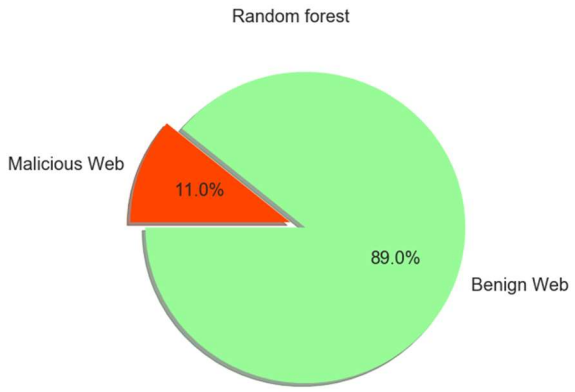
Penelitian ini dataset yang kami gunakan berisikan 1000 web serta terdapat dua jenis variabel yang digunakan terdiri dari 7 yakni variabel prediksi dan 1 variabel respon.

Variabel prediksi terdiri dari *URL LENGTH, NUMBER SPECIAL CHARACTERS, REMOTE IPS, REMOTE APP PACKETS, SOURCE APPBYTES, REMOTE APP BYTES* dan *APP PACKET. URL LENGTH* adalah banyaknya karakter yang terdapat didalam url, *NUMBER SPECIAL CHARACTERS* adalah jumlah karakter yang bersifat khusus yang terdapat didalam url misalnya :"/","%" , *REMOTE IPS*

ISBN : 978-979-587-813-1
Vol.4 No.1

adalah variabel yang memiliki jumlah total IP yang terhubung ke honeypot, *REMOTE APP PACKETS* adalah paket yang diterima dari server, *SOURCE APP BYTES* adalah jumlah byte yang di transfer. *REMOTE APP BYTES* adalah jumlah byte yang ditransfer dari server. *APP PACKET* adalah jumlah total paket IP yang dihasilkan selama komunikasi antara honeypot dan server.

Variabel respon digunakan adalah *class Type* yang bertipe integer yakni 1 dan 0, jika memenuhi respon bernilai 1 maka dapat dikatakan web tersebut *malicious web* dan jika memenuhi respon bernilai 0 maka web tersebut dapat dikatakan *benign web*.



Gambar .1.Data jumlah benign web dan malicious web

Dari gambar 1 dapat dilihat bahwa komposisi data *benign web* adalah 89,0% sedangkan komposisi *malicious web* adalah 11,00% .Sebelum data di pakai, data di bagi menjadi dua bagian, 70 % data *training* dan 30% data *testing*.

B. RANDOM FOREST

Pada penelitian ini kami menggunakan Metode *Random Forest* (RF) untuk mengidentifikasi *malicious web.* *Random forest* merupakan salah satu metode yang digunakan untuk klasifikasi dengan membangun banyak pohon klasifikasi. RF dapat meningkatkan akurasi karena adanya pemilihan secara acak dalam membangkitkan simpul anak untuk setiap node (simpul diatasnya) dan diakumulasikan hasil klasifikasi dari setiap pohon, kemudian dipilih hasil klasifikasi yang paling banyak muncul [7].

Pada operator *Random Forest* menghasilkan satu set pohon acak, kelas yang dihasilkan dari proses klasifikasi dipilih dari kelas yang paling banyak (modus) yang dihasilkan oleh *tree* acak yang ada. Banyak pohon ditumbuhkan dalam metode *Random Forest*, sehingga terbentuk *forest* yang akan dianalisis dan diteliti. Pada gugus data yang terdiri atas *n* amatan dan *p* peubah penjelas, *Random Forest* dilakukan dengan cara melakukan tahapan *bootstrap,* tahapan *bootsrap* yaitu penarikan contoh acak berukuran *n* dengan pemulihan pada gugus data kemudian dengan menggunakan contoh *bootstrap*, *tree* dibangun sampai mencapai ukuran maksimum (tanpa pemangkasan). Pada setiap simpul, pemilihan pemilah dilakukan dengan memilih *m* peubah penjelas secara acak, dimana $m << p$, lalu pemilah terbaik dipilih berdasarkan *m* peubah penjelas tersebut dimana tahapan ini disebut dengan tahapan *random feature selection*, lalu Ulangi langkah 1 dan 2 sebanyak *k* kali, sehingga terbentuk sebuah *forest* yang terdiri atas *k tree* [8] .

Tahapan pembuatan model klasifikasi menggunakan algoritma *Random Forest* dilakukan setelah membuat pemodelan data latih menggunakan *package randomForest* di R. Pembentukan *tree* pada algoritma *Random Forest* dilakukan dengan cara melakukan *training* pada sampel. Variabel yang digunakan untuk split diambil secara acak dan klasifikasi dijalankan setelah semua *tree* terbentuk. Penentuan klasifikasi pada *Random Forest* ini diambil berdasarkan vote dari masing-masing *tree* dan *vote* terbanyak yang menjadi pemenang. Pada pembentukan *Random Forest* menggunkan nilai *Gini Index* untuk menentukan split yang akan dijadikan *root/node*[7]. Berikut ini adalah persamaan untuk mencari nilai *Gini*

$$\mathbf{Gini}\ (\mathbf{S}) = \mathbf{1} - \sum_{i=1}^{k} \boldsymbol{pi}^{2} \quad (1)$$

Dimana pi adalah probabilitas dari S milik *class i*. Setelah menghitung nilai *Gini* (S), langkah berikutnya adalah menghitung nilai *GiniGain*.

$$\mathbf{GiniGain}\ (\mathbf{S}) = \mathbf{Gini}\ (\mathbf{S}) - \mathbf{Gini}\ (\mathbf{A}, \mathbf{S}) = \mathbf{Gini}\ (\mathbf{S}) - \sum_{\mathbf{i=1}}^{\mathbf{n}} \frac{|\mathbf{S_i}|}{|\mathbf{S}|}\ \mathbf{Gini}\ (\mathbf{S_i}) \quad (2)$$

Dimana Si adalah partisi dari S yang disebabkan oleh atribut A [11]

C. MATRIKS CONFUSION

Tahap selanjutnya pada penelitian ini adalah tahap perhitungan menggunakan matriks *confusion*. Matriks *confusion* adalah suatu tabel yang sering digunakan untuk menggambarkan kinerja model klasifikasi pada suatu set data *testing* .

TABEL I.*Matriks Confusion*

ISBN : 978-979-587-813-1
Vol.4 No.1

TP (*True Positive*) adalah kasus di mana kita memprediksi ya dan nilai aktualnya benar, FN (*False Negative*) adalah kasusnya diprediksi tidak dan nilai aktualnya benar, FP (*False Positives*) adalah kasus yang kami perkirakan ya dan nilai aktualnya adalah salah, TN (*True Negatives*) adalah kasus di mana kami memprediksi tidak dan nilai salah.

Nilai matriks *confusion* pada penelitian ini dapat dilihat pada table (II).

D.PERHITUNGAN AKURASI

Perhitungan akurasi dilakukan setelah proses klasifikasi selesai dilakukan. Perhitungan ini berfungsi menunjukkan tingkat kebenaran pengkalisifikasian data terhadap data yang sebenarnya. Perhitungan akurasi akan dilakukan dengan menggunakan persamaan (4).

$$\textbf{Akurasi} = \frac{\textbf{TP+TN}}{\textbf{TP+TN+FP+FN}} \textbf{x 100\%} \quad (3)$$

Untuk mengetahui nilai TP, TN, FP dan FN pada penelitian ini dapat dilihat pada tabel (II).

$$Akurasi = \frac{85+0}{85+6+6+0} x\ 100\% = 0{,}94$$

Jadi, didapatkan hasil akurasi pada penelitian ini adalah 94%

Selain dapat menghitung akurasi, kita juga dapat menghitung nilai eror dalam hal prediksi menggunakan persamaan (6).

$$\textbf{Eror} = \frac{\textbf{FP+FN}}{\textbf{TP+TN+FP+FN}} \quad (4)$$

E.KURVA ROC

Untuk mengetahui penelitian ini baik atau tidak kami menggunakan kurva ROC. Kurva ROC adalah sebuah penghitungan untuk menilai akurasi dari sebuah prediksi. Sebuah prediksi dibuat sebelum nilai dari entitas yang diprediksi tersebut diketahui. Oleh karena itu, diperlukan sebuah metode untuk mengevaluasi akurasi dari berbagai prediksi tersebut [12].

Nilai yang diplot adalah nilai TPR dan FPR, dimana masing-masing nilai dapat dihitung dengan menggunakan persamaan (5) dan persamaan (6).

$$\textbf{TPR} = \frac{\textbf{TP}}{\textbf{TP+FP}} \quad (5)$$

$$\textbf{FPR} = \frac{\textbf{FP}}{\textbf{TP+FP}} \quad (6)$$

		Predicted Class	
		P	N
Actual class	P	*True Positives* (TP)	*False Negatives*(FN)
	N	*False Positives* (FP)	*True Negatives* (TN)

Pada kurva ROC apabila kurva semakin bertambah besar terhadap luas daerahnya maka hasil uji tersebut semakin baik dan sebaliknya apabila kurva semakin bertambah kecil terhadap luas daerahnya maka hasil uji semakin buruk.

III. HASIL DAN PEMBAHASAN

Berdasarkan hasil penelitian yang telah kami lakukan maka didapatkan hasil seperti tabel (II).

TABEL II. HASIL BERUPA NILAI *PRECISION, RECALL, F1-SCORE* DAN *SUPPORT.*

Variable	Precision	Recall	F1-Score	Support
0	0.93	1.00	0.97	85
1	1.00	0.50	0.67	12
Avg/total	0.94	0.94	0.93	97

Dari tabel II dapat dilihat hasil tahap prediksi yaitu nilai *precision* adalah 94%, nilai *recall* adalah 94%, nilai *F1-score* adalah 93%, dan nilai *support* nya adalah 97.

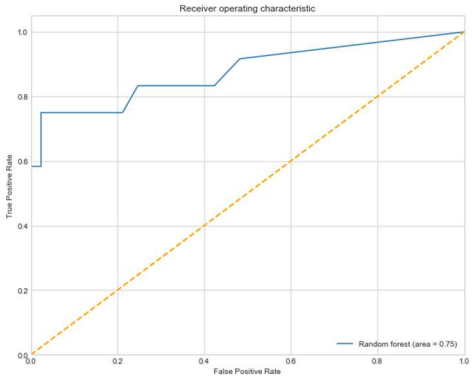
Kemudian kami melakukan perhitungan matriks confusion serta perhitungan nilai akurasi .

TABEL III.NILAI *MATRIKS CONFUSION* DARI HASIL PENELITIAN

	Predicted Class	
Actual	85	0
class	6	6

Hasil nilai matriks *confusion* pada penelitian ini dapat dilihat pada tabel III. Selanjutnya kami melakukan penelitian menggunakan matiks *confusion* untuk lebih memperjelas hasil akurasi data dengan menggunakan data pada tabel III. Data pada tabel III dapat dihitung tingkat akurasinya dengan menggunakan persamaan (3) dan nilai eror dengan menggunakan persamaan (4). Maka didapatkan tingkat akurasi sebesar 94% sedangkan nilai eror adalah 6%.

ISBN : 978-979-587-813-1
Vol.4 No.1



Gambar 2. Visualisasi hasil penelitian menggunakan kurva ROC

Berdasarkan gambar (2) kurva ROC diatas dapat dilihat bahwa hasil penelitian mengidentifikasi *malicious web* menggunakan metode *Random Forest* hasil ujinya baik karena kurva semakin besar terhadap luas daerahnya.

IV. KESIMPULAN

Pada penelitian mengidentifikasi *Malicious Web* menggunakan *Random Forest* didapatkanan bahwa akurasi yang didapatkan pada penelitian ini sebesar 94% sedangkan nilai eror sebesar 6%, nilai *precision* adalah 94%, nilai *recall* adalah 94%, nilai *F1-score* adalah 93%, dan nilai *support* nya adalah 97 serta hasil ujinya baik karena jarak pada kurva ROC besar terhadap luas daerahnya.

REFERENSI

[1] D. Canali, M. Cova, G. Vigna, and C. Kruegel, "Prophiler: A Fast Filter for the Large-Scale Detection of Malicious Web Pages," *WWW '11 Proc. 20th Int. Conf. World wide web*, pp. 197–206, 2011.

[2] H. Choi, B. B. Zhu, and H. Lee, "Detecting malicious web links and identifying their attack types," *WebApps*, p. 11, 2011.

[3] H. Mekky, R. Torres, Z.-L. Zhang, S. Saha, and A. Nucci, "Detecting malicious HTTP redirections using trees of user browsing activity. BT - 2014 IEEE Conference on Computer Communications, INFOCOM 2014, Toronto, Canada, April 27 - May 2, 2014," pp. 1159–1167, 2014.

[4] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Learning to detect malicious urls," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, p. 30, 2011.

[5] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond Blacklists : Learning to Detect Malicious Web Sites from Suspicious URLs," pp. 1245–1253, 2009.

[6] G. A. Sandag, J. Leopold, and V. F. Ong, "Klasifikasi Malicious Websites Menggunakan Algoritma K-NN Berdasarkan Application Layers dan Network Characteristics Malicious Websites Classification Using K-NN Algorithm Based on Application Layers and Network Characteristics," *Cogito Smart J.*, vol. 4, no. 1, pp. 37–45, 2018.

[7] A. Zainal, M. A. Maarof, S. M. Shamsuddin, and A. Abraham, "Ensemble of one-class classifiers for network intrusion detection system," *Proc. - 4th Int. Symp. Inf. Assur. Secur. IAS 2008*, vol. 4, pp. 180–185, 2008.

[8] M. ROSAS PÉREZ, *Propuesta de un proceso de recolección de residuos electrónicos para motivar la participación en poblaciones definidas*. 2012.

[9] T. Dalgleish *et al.*, *Classification And Regression Trees*, vol. 136, no. 1. 2007.

[10] D. Fransiska Amalia Kurniawan, "Analisis dan Implementasi Random Forest dan Regression Tree (CART) Untuk Klasifikasi pada Misuse Intrussion Detection System," *Fak. Tek. Inform.*, no. Data Mining, pp. 1–7, 2011.

[11] D. I. Komputer, F. Matematika, D. A. N. Ilmu, and P. Alam, "Klasifikasi kemunculan titik panas pada lahan gambut di sumatera dan kalimantan dengan menggunakan algoritme pohon keputusan c5.0 dhita aprita," 2016.

[12] R. Hidayat and I. Primasari, "Metodologi Penelitian Psikodiagnostika," *Bul. Psikol.*, vol. 19, no. 2, pp. 81–92, 2011.

[13] A. Hanley, J. Mcneil, and D. Ph, "under a Receiver Characteristic," pp. 29–36.