BAB 5
PENUTUP

5.1 KESIMPULAN

Berdasarkan implementasi deteksi dan pencegahan serangan DDoS pada *server* menggunakan Snort dengan notifikasi *email*, maka dapat diperoleh beberapa kesimpulan sebagai berikut:

1. Hasil implementasi deteksi dan pencegahan adalah bahwa Snort telah berhasil mendeteksi baik itu lalu lintas DDoS dan Normal melalui *custome rules* yang menandakan bahwa *rules* sesuai dengan lalu lintas yang terjadi. Pada *pfSense* juga berhasil memblokir IP melalui tiga tindakan pemblokiran IP yaitu melalui *block snort2c host, block ipv4 link-local, dan block traffic from port 0*. Jadi kombinasi Snort dengan fitur perlindungan lainnya di *pfSense* dapat membantu meningkatkan keamanan jaringan dan melindungi dari serangan DDoS khususnya TCP *Flood* dan UDP *Flood*.
2. Hasil implementasi notifikasi adalah bahwa notifikasi dibangun melalui paket tambahan *pfSense* yaitu *EmailReport*, dengan sistem notifikasi secara *periodic* dan berhasil mengirimkan pesan *log* Snort dengan frekuensi pengiriman laporan harian pada pukul 22.10.
3. Pada klasifikasi *log* Snort, menghasilkan nilai metrik evaluasi yang tinggi seperti metrik *accuracy* 98,6%, *precision* 99,4%, *recall* 98,7%, dan *specificity* 98,4%. Selain itu, pada metrik *f1-score* juga tinggi 99%, menunjukkan bahwa model dapat menjaga keseimbangan antara *precision* dan *recall*. Meskipun FPR memiliki nilai yang rendah, hal ini bisa menjadi area yang perlu diperhatikan untuk mengurangi kesalahan klasifikasi Normal sebagai DDoS. Secara keseluruhan menunjukkan akurasi dan efektivitas Snort sangat baik dalam mendeteksi DDoS dan Normal.
4. Pada *detection latency* skenario DDoS menunjukkan nilai standar deviasi sebesar 0,51 detik, artinya *respons* sistem terhadap UDP *Flood* dapat bervariasi lebih luas dibandingkan dengan TCP *Flood* sebesar 0,40 detik. Sedangkan pada skenario Normal menunjukkan nilai standar deviasi yang sama-sama 0 detik,

artinya tidak ada variasi respons. Kmudian, pada CPU *Usage* skenario DDoS menunjukkan nilai standar deviasi sebesar (2,5%), artinya tingkat variasi penggunaan CPU TCP *Flood* relatif kecil dibandingkan dengan UDP *Flood* sebesar (12,3%). Sedangkan pada skenario Normal menunjukkan nilai standar deviasi penggunaan CPU sebesar 1,2% (TCP Normal) dan 1,0% (UDP Normal), artinya variasi penggunaan CPU relatif serupa atau tidak jauh berbeda satu sama lain. Variasi *detection latency* penggunaan CPU dapat disebabkan oleh faktor karakteristik pengiriman paket.

5.2 SARAN

Berdasarkan penelitian deteksi dan pencegahan serangan DDoS pada *server* menggunakan Snort dengan notifikasi *email*, maka dapat diperoleh beberapa saran untuk penelitian selanjutnya, sebagai berikut:

1. Penelitian selanjutnya dapat melakukan pengembangan lebih komprehensif pada Snort dan *pfSense*, seperti variasi *rules* baru dan menggunakan kombinasi sistem keamanan yang lain seperti *pfblocker* dalam menangkal atau mencegahan serangan DDoS yang memiliki sistem pengiriman paket berkecepatan tinggi.
2. Penelitian selanjutnya dapat menggunakan fitur atau jenis media notifikasi lain yang dimiliki *pfSense*, seperti *pushover*.
3. Penelitian selanjutnya dapat menggunakan jumlah variasi *fold* yang berbeda seperti 2, 3, 5 maupun 20. Kemudian analisis penggunaan *fold* jika terlalu besar atau sebaliknya dengan ukuran dataset yang dimiliki.
4. Penelitian selanjutnya dapat menggunakan algoritma yang lain seperti *random forest*, *naïve baiyes*, dan regresi logistik. Kemudian dapat melakukan perbandingan kinerja antara algoritma tersebut dalam klasifikasi *log* Snort.