DETEKSI SERANGAN *MAN IN THE MIDDLE* PADA PROTOKOL *MODBUS* DI JARINGAN *SMART GRID* MENGGUNAKAN ALGORITMA *RANDOM FOREST*

SKRIPSI

Oleh :
SHOLIKIN
NIM. 200605110119

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024

DETEKSI SERANGAN *MAN IN THE MIDDLE* PADA PROTOKOL *MODBUS* DI JARINGAN *SMART GRID* MENGGUNAKAN ALGORITMA *RANDOM FOREST*

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
SHOLIKIN
NIM. 200605110119

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2024

HALAMAN PERSETUJUAN

DETEKSI SERANGAN *MAN IN THE MIDDLE* PADA PROTOKOL *MODBUS* DI JARINGAN *SMART GRID* MENGGUNAKAN ALGORITMA *RANDOM FOREST*

SKRIPSI

Oleh :
SHOLIKIN
NIM. 200605110119

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 15 November 2024

Pembimbing I,	Pembimbing II,
Johan Ericka Wahyu Prakasa, M.Kom	Hani Nurhayati, M.T
NIP. 19831213 201903 1 004	NIP. 19780625 200801 2 006

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Ir. Fachrul Kurniawan, M.MT, IPU
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

DETEKSI SERANGAN *MAN IN THE MIDDLE* PADA PROTOKOL *MODBUS* DI JARINGAN *SMART GRID* MENGGUNAKAN ALGORITMA *RANDOM FOREST*

SKRIPSI

Oleh :
SHOLIKIN
NIM. 200605110119

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 6 Desember 2024

Susunan Dewan Penguji

Ketua Penguji	: Dr. M. Amin Hariyadi, M.T NIP. 19670118 200501 1 001	( )
Anggota Penguji I	: Ajib Hanani, M.T NIP. 19840731 202321 1 013	( )
Anggota Penguji II	: Johan Ericka Wahyu Prakasa, M.Kom NIP. 19831213 201903 1 004	( )
Anggota Penguji III	: Hani Nurhayati, M.T NIP. 19780625 200801 2 006	( )

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Ir. Fachrul Kurniawan, M.MT, IPU
NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama	: Sholikin
NIM	: 200605110119
Fakultas / Program Studi	: Sains dan Teknologi / Teknik Informatika
Judul Skripsi	: DETEKSI SERANGAN *MAN IN THE MIDDLE* PADA PROTOKOL MODBUS DI JARINGAN *SMART GRID* MENGGUNAKAN ALGORITMA *RANDOM FOREST*.

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 6 Desember 2024
Yang membuat pernyataan,

Sholikin
NIM.200605110119

MOTTO

Ilmu Amaliah Amal Ilmiah

"Sesungguhnya sesudah kesulitan itu ada kemudahan. Maka apabila kamu telah selesai (dari sesuatu urusan), kerjakanlah dengan sungguh-sungguh (urusan) yang lain."

(*QS. Al-Insyirah: 6-7*)

HALAMAN PERSEMBAHAN

Alhamdulillah Wa Syukurillah segala Puji dan Syukur kehadirat Allah *Subhanahu Wa Ta'ala* atas limpahan rahmat, taufiq, hidayah, dan inayah-Nya kepada penulis beserta keluarga sehingga penulis dapat menyelesaikan tugas akhir ini. Dengan tulus dan rasa bersyukur sebagai penghargaan yang sebesar-besarnya, Penulis menyampaikan persembahan ini sebagai bentuk dedikasi dan bukti tanggung jawab dengan selesainya skripsi ini kepada kedua orang tua penulis dan diri penulis. Terima kasih sudah selalu mendoakan penulis di setiap waktunya serta senantiasa memberikan dukungan penuh dengan sangat baik dan tulus. Terima kasih atas kesabaran, kekuatan dan kepercayaan yang selalu kalian berikan sehingga menjadi sumber semangat untuk menuntut ilmu dan menyelesaikan skripsi ini. Terima kasih juga penulis ucapkan untuk diri sendiri yang sudah sabar, kuat, dan semangat dalam menuntut ilmu dan bertanggung-jawab untuk menyelesaian pilihan yang telah dimulai. Semoga karya ini menjadi karya yang bermanfaat dunia dan akhirat serta menjadi keberkahan untuk penulis, keluarga penulis dan semuanya. Amiin….

KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarakatuh.

Alhamdulillah Wa Syukurillah. Dengan penuh rasa syukur, penulis panjatkan puja dan puji kepada Allah *Subhanahu Wa Ta'ala* atas limpahan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul "**Deteksi Serangan *Man In The Middle* pada Protokol Modbus di Jaringan *Smart Grid* Menggunakan Algoritma *Random Forest***" dengan lancar. Melalui kesempatan ini, penulis menyampaikan rasa terimakasih kepada semua pihak yang telah mendukung terselesaikannya skripsi ini. Terutama kepada kedua orang tua penulis yang selalu memberikan dukungan moral dan material serta selalu mendoakan penulis kapan pun dan diamana pun. Selain itu penulis juga mengucapkan terimaksih kepada :

1. Prof. Dr. H. M. Zainuddin, M.A., selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Harini, M.Si., selaku Dekan Fakultas Sains dan Teknologi Universitas UIN Maulana Malik Ibrahim Malang.
3. Dr. Ir. Fachrul Kurniawan, M.MT, IPU, selaku Ketua Program studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang
4. Fatchurrohman, M.Kom selaku Dosen Wali yang telah memberikan arahan dalam proses perkuliahan, penulis ucapkan terima kasih sebanyak-banyaknya.
5. Johan Ericka Wahyu Prakasa, M.Kom selaku Dosen Pembimbing 1 skripsi, penulis ucapkan terima kasih sebanyak-banyaknya karena selalu memberikan

masukan dan juga ilmu yang beliau miliki kepada penulis, serta memberikan dukungan, bimbingan dan bantuan dalam terwujudnya penyusunan skripsi ini.

6. Hani Nurhayati, M.T selaku Dosen Pembimbing 2 yang telah memberikan bimbingan dan arahan pada penulisan karya tulis skripsi ini.
7. Dr. M. Amin Hariyadi, M.T. selaku Dosen Penguji 1 dan Ajib Hanani, M.T selaku Dosen Penguji 2 yang telah memberikan arahan dalam menyelesaikan skripsi ini.
8. Segenap Dosen, Laboran, Admin dan jajaran pada Program Studi Teknik Informatika yang telah memberikan bimbingan dan bantuan selama studi.
9. Ayah dan Ibu penulis yang selalu memberi doa, dukungan dan perhatian serta selalu memberikan yang terbaik untuk kelancaran penulis dalam segala hal.
10. Seluruh Guru, para Ustadz dan para Kyai yang telah memberikan doa dan dukungannya kepada penulis.
11. Seluruh keluarga besar Saudara Teknik Informatika UIN Maulana Malik Ibrahim Malang khususnya Angkatan 2020 “INTEGER”, serta teman-teman dekat penulis, terimakasih telah memberikan *support*, semangat, motivasi dan bantuannya kepada penulis.
12. Diri sendiri yang sudah sabar, kuat, dan semangat dalam menuntu ilmu. Serta seluruh pihak yang telah terlibat secara langsung maupun tidak langsung dalam proses penyusunan skripsi ini.

Malang, 22 November 2024

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Sholikin. 2024. **DETEKSI SERANGAN *MAN IN THE MIDDLE* PADA PROTOKOL MODBUS DI JARINGAN *SMART GRID* MENGGUNAKAN ALGORITMA *RANDOM FOREST*.** Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Johan Ericka Wahyu Prakasa, M.Kom (II) Hani Nurhayati, M.T.

.

Kata Kunci: Deteksi Serangan, *Man in The Middle, Random Forest, Smart Grid*.

Smart Grid merupakan jaringan listrik yang terintegrasi dengan teknologi informasi untuk mengelola energi listrik agar lebih efisien. Modbus adalah protokol yang digunakan pada *Smart Grid* untuk pertukaran data dengan cepat namun rentan terhadap serangan jaringan, khususnya *Man in The Middle* (MITM) yang dapat menyebabkan kerugian yang besar sehingga perlu adanya deteksi serangan pada protokol Modbus. *Random Forest* dipilih karena dapat melakukan klasifikasi data dalam jumlah besar dalam waktu yang relatif singkat dan merupakan *ensamble learning* yang dapat mencegah *overfitting* dan meningkatkan akurasi pada proses klasifikasi data. Data yang digunakan sebanyak 5583 baris dengan 6 fitur dan 4 kelas, kemudian dilakukan *preprocessing* yaitu *missing value*, pelabelan kelas dan *split data*. Pengujian dilakukan dengan tiga uji coba, di mana setiap uji coba mengunakan kelas normal dan 1 kelas serangan. Kemudian dilakukan *split data* pada tiap uji coba. *Split data* dilakukan dengan membagi menjadi 5 model dengan perbandingan data latih:data uji pada model 1 adalah (50%:50%), model 2 adalah (60%:40%), model 3 adalah (70%:30%), model 4 adalah (80%:20%) dan model 5 adalah (90%:10%). Setiap model dilakukan pengaturan *hyperparameter*, yaitu *n_estimators* dan *max_features*. Pada uji coba yang dilakukan didapatkan hasil terbaik pada uji coba 1 pada proporsi data latih:data uji 60%:40% hingga 90%:10% yang menghasilkan akurasi 100%, presisi 100%, *recall* 100% dan *f1-score* sebesar 100%. Hasil ini membuktikan bahwa algoritma *Random Forest* efektif untuk mendeteksi serangan MITM pada jaringan *Smart Grid*.

ABSTRACT

Sholikin. 2024. **DETECTION OF MAN IN THE MIDDLE ATTACK ON MODBUS PROTOCOL IN SMART GRID NETWORK USING RANDOM FOREST ALGORYTHM.** Thesis. Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University, Maulana Malik Ibrahim Malang. Promoter: (I) Johan Ericka Wahyu Prakasa, M.Kom (II) Hani Nurhayati, M.T.

Smart Grid is an electrical network integrated with information technology to manage electrical energy more efficiently. Modbus is a protocol used in Smart Grid to exchange data quickly but is vulnerable to network attacks, especially Man in The Middle (MITM) which can cause huge losses so it is necessary to detect attacks on the Modbus protocol. Random Forest was chosen because it can classify large amounts of data in a relatively short time and is ensemble learning that can prevent overfitting and improve accuracy in the data classification process. The data used was 5583 rows with 6 features and 4 classes, then preprocessing was carried out, namely missing values, class labeling and split data. Testing was carried out with three trials, where each trial used a normal class and 1 attack class. Then split data is performed on each trial. Split data is done by dividing into 5 models with the ratio of training data:testing data in model 1 is (50%:50%), model 2 is (60%:40%), model 3 is (70%:30%), model 4 is (80%:20%) and model 5 is (90%:10%). Each model is set hyperparameters, namely n_estimators and max_features. In the trials conducted, the best results were obtained in trial 1 at the proportion of training:testing data 60%:40% to 90%:10% which resulted in 100% accuracy, 100% precision, 100% recall and f1-score of 100%. These results prove that the Random Forest algorithm is effective for detecting MITM attacks on Smart Grid networks.

Key words: Attack Detection, Man in The Middle, Random Forest, Smart Grid.

مستخلصالبحث

صَالِحِيْن ٢٠٢٤. كشف هجوم الرجل في الوسط على بروتوكول مودبوس في شبكات الشبكة الذكية باستخدام خوارزمية الغابة العشوائية. أطروحة. قسم هندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم مالانج الإسلامية الحكومية. المشرف: (١) يوهان إيريكا واهيو براكاسا، م. كوم (٢) هاني نورهاياتي، م. ت.

الكلمات المفتاحية: الكشف عن الهجمات، رجل في الوسط، الغابة العشوائية، الشبكة الذكية.

الشبكة الذكية هي شبكة كهربائية مدمجة مع تكنولوجيا المعلومات لإدارة الطاقة الكهربائية بكفاءة أكبر. مودبوس هو بروتوكول يستخدم في الشبكة الذكية لتبادل البيانات بسرعة ولكنه عرضة لهجمات الشبكة، خاصةً الرجل في الوسط (MITM) الذي يمكن أن يتسبب في خسائر فادحة، لذا من الضروري الكشف عن الهجمات على بروتوكول مودبوس. تم اختيار الغابة العشوائية لأنها قادرة على تصنيف كميات كبيرة من البيانات في وقت قصير نسبيًا وهي عبارة عن مجموعة تعلم يمكن أن تمنع الإفراط في التكييف وتحسن الدقة في عملية تصنيف البيانات. البيانات المستخدمة عبارة عن 5583 صفًا مع 6 ميزات و4 فئات، ثم تمت المعالجة المسبقة، وهي القيم المفقودة وتسمية الفئات وتقسيم البيانات. تم إجراء الاختبار بثلاث تجارب، حيث استخدمت كل تجربة فئة عادية وفئة هجوم واحدة. ثم يتم إجراء تقسيم البيانات على كل تجربة. يتم تقسيم البيانات المجزأة من خلال تقسيم البيانات إلى 5 نماذج بنسبة بيانات التدريب: بيانات الاختبار في النموذج 1 هي (50%:50%)، والنموذج 2 (60%:40%)، والنموذج 3 (70%:30%)، والنموذج 4 (80%:20%)، والنموذج 5 (90%:10%). تم تعيين كل نموذج بمحددات فائقة، وهي n_estimators و max_features. في التجارب التي تم إجراؤها، تم الحصول على أفضل النتائج في التجربة 1 عند نسبة بيانات التدريب:بيانات الاختبار 60%:40% إلى 90%:10%، والتي أسفرت عن دقة 100% ودقة 100% واسترجاع 100% ودرجة 100% f1. وتثبت هذه النتائج أن خوارزمية الغابة العشوائية فعالة في الكشف عن هجوم رجل في الوسط الإلكتروني على شبكات الشبكة الذكية.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Jaringan listrik pintar atau *Smart Grid* merupakan evolusi dari jaringan listrik konvensional yang didalamnya terdapat integrasi dengan teknologi informasi (Sinaga et al., 2021). *Smart Grid* mampu mengintegrasikan berbagai aktivitas dari semua pengguna energi, mulai dari pembangkit listrik hingga konsumen, dengan tujuan meningkatkan efisiensi, pertumbuhan ekonomi, dan keamanan pasokan energi listrik (Sinaga et al., 2021). *Smart Grid* memanfaatkan teknologi informasi dan komunikasi untuk mengawasi dan mengendalikan jaringan dengan lebih efisien dan efektif, memastikan pasokan listrik dapat disesuaikan dengan penggunaan, sehingga meningkatkan integrasi sumber daya terbarukan ke dalam jaringan. Sistem ini memanfaatkan sensor, *smart meter*, perangkat lunak analitik, dan jaringan komunikasi digital untuk memantau dan mengendalikan jaringan (A. Ghosh & Kole, 2021).

Pada *Smart Grid, Distributed Energy Resources* (DER) dan *Private Area Network* (PAN) memainkan peran yang penting. DER adalah sumber energi terdistribusi, seperti sistem penyimpanan energi, panel surya, dan turbin angin yang terhubung ke jaringan listrik. DER juga memiliki kemampuan untuk menghasilkan energi secara terdesentralisasi dan dapat diintegrasikan ke dalam jaringan listrik melalui (PAN) (Elrawy et al., 2023). Sedangkan *Private Area Network* (PAN) merupakan jaringan lokal yang terdiri dari perangkat-perangkat

seperti *smart meter*, sensor, pengontrol lokal, dan peralatan listrik yang terhubung ke jaringan *Smart Grid.* Dengan menggunakan PAN, penggunaan energi terbarukan dari DER dapat dimonitor, dikendalikan, dan diintegrasikan ke dalam jaringan *Smart Grid* dengan lebih efisien dan efektif (Elrawy et al., 2023).

Integrasi DERS dan PAN dalam *Smart Grid* memungkinkan pengelolaan energi listrik secara lebih terdistribusi dan responsif, sehingga meningkatkan efisiensi dan keandalan jaringan listrik secara keseluruhan. Dengan adanya *Smart Grid*, jaringan listrik dapat lebih responsif terhadap perubahan permintaan, mengurangi kehilangan energi, meningkatkan integrasi sumber energi terbarukan, serta meminimalkan dampak lingkungan. Selain itu, *Smart Grid* juga memungkinkan penggunaan teknologi baru seperti mobil listrik dan penyimpanan energi, serta memberikan peluang bagi inovasi dalam manajemen energi (Sinaga et al., 2021).

Dengan banyaknya jumlah entitas yang terhubung pada *Smart Grid*, maka *Smart Grid* memiliki banyak jenis protokol. Salah satu protokol komunikasi yang umum digunakan pada *Smart Grid* adalah Modbus TCP/IP. Protokol ini digunakan untuk mentransmisikan data antara perangkat dalam jaringan, seperti pengontrol lokal di lokasi DER dan peralatan listrik di PAN (Elrawy et al., 2023). Modbus TCP/IP memungkinkan pertukaran data yang handal dan efisien, serta memfasilitasi integrasi yang lebih baik dari sumber energi terdistribusi. Dengan menggunakan protokol ini, data operasional dapat dikirimkan antara perangkat di lapangan dan pusat kontrol dengan cepat dan akurat, sehingga memungkinkan pengelolaan jaringan yang lebih efisien dan responsif. Selain itu, Modbus TCP/IP

juga memungkinkan integrasi yang lebih baik antara perangkat berbasis *legacy* dan sistem-sistem yang lebih modern dalam *Smart Grid* (A. Ghosh & Kole, 2021).

Protokol Modbus menjadi salah satu komponen dalam membangun infrastruktur pada *Smart Grid* agar menjadi handal dan efisien. Namun, protokol ini seringkali tidak memiliki mekanisme kerahasiaan informasi. Sehingga membuatnya rentan terhadap serangan jaringan. Salah satu serangan yang paling umum ditujukan kepada *Smart Grid*, terutama terhadap protokol TCP/IP Modbus, adalah serangan *Man in The Middle* (MITM) (Elrawy et al., 2023).

Serangan *Man in The Middle* (MITM) merupakan serangan yang mengintersepsi komunikasi antara 2 perangkat (Farhan, 2021). Serangan ini terjadi ketika seorang penyerang berhasil menyusup ke dalam komunikasi antara dua perangkat atau sistem dan memanipulasi data yang dikirimkan di antara keduanya tanpa sepengetahuan mereka (Septiano, 2021). Hal ini tentunya akan sangat merugikan karena serangan MITM terhadap protokol Modbus TCP/IP pada *Smart Grid* dapat menyebabkan pencurian data, pemalsuan data, atau bahkan gangguan terhadap operasi jaringan secara keseluruhan. Jika hal ini terjadi maka akan mengakibatkan kerugian & kerusakan yang besar.

Random Forest adalah metode *ensamble learning* yang mana pada dasarnya *Random Forest* menggabungkan beberapa model pohon keputusan untuk meningkatkan kinerja dan hasil prediksinya (Septiano, 2021). Metode ini biasa digunakan untuk klasifikasi dengan membangun beberapa pohon keputusan untuk menyelesaikan masalah. *Random Forest* dapat meningkatkan tingkat akurasi

karena dalam prosesnya terdapat pemilihan data menggunakan teknik *boostrap* yaitu pengambilan secara acak dalam pembentukan *node* anak untuk setiap *node* di atasnya, serta akumulasi hasil klasifikasi dari setiap pohon (Septiano, 2021). Hasil klasifikasi yang paling sering terjadi akan dipilih sebagai hasil akhir. Metode ini biasanya diterapkan untuk mengelola data dengan ukuran atau format yang bervariasi. Dalam proses pelatihan, *Random Forest* menggunakan data lalu lintas jaringan yang telah ada. Sehingga, penambahan data secara terus-menerus akan secara signifikan meningkatkan tingkat akurasi yang diperoleh.

Penelitian tentang deteksi serangan di protokol Modbus menggunakan *Random Forest* dilakukan oleh (T. Ghosh et al., 2023) untuk mengidentifikasi serangan *Man in The Middle*, pada sistem kontrol industri. Penelitian ini menggunakan 2 pembagian data yang berbeda yaitu pembagian 50–50 (50% data non-serangan dan 50% serangan) dan pembagian 80-20 (80% data non-serangan dan 20% serangan). Penelitian ini juga menggunakan kombinasi data pelatihan dan pengujian dimana untuk kombinasi pertama terdiri dari 60% *training* dan 40% *testing.* Untuk kombinasi lainnya menggunakan 75% *training* dan 25% *testing.* Hasil akurasi terbaik yang didapatkan dari *Random Forrest* dengan pembagian data *training* dan *testing* sebesar 75% dan 25% dengan perbandingan 80-20 menghasilkan nilai akurasi terbaik sebesar 86,41%.

Pada QS *Ali Imran* ayat 103 relevan dengan penelitian ini

وَاعْتَصِمُوْا بِحَبْلِ اللّٰهِ جَمِيْعًا وَّلَا تَفَرَّقُوْاۚ وَاذْكُرُوْا نِعْمَتَ اللّٰهِ عَلَيْكُمْ اِذْ كُنْتُمْ اَعْدَاۤءً فَاَلَّفَ بَيْنَ قُلُوْبِكُمْ
فَاَصْبَحْتُمْ بِنِعْمَتِهٖٓ اِخْوَانًاۚ وَكُنْتُمْ عَلٰى شَفَا حُفْرَةٍ مِّنَ النَّارِ فَاَنْقَذَكُمْ مِّنْهَاۗ كَذٰلِكَ يُبَيِّنُ اللّٰهُ لَكُمْ اٰيٰتِهٖ لَعَلَّكُمْ
تَهْتَدُوْنَ

"Berpegangteguhlah kamu semuanya pada tali (agama) Allah, janganlah bercerai berai, dan ingatlah nikmat Allah kepadamu ketika kamu dahulu bermusuhan, lalu Allah mempersatukan hatimu sehingga dengan karunia-Nya kamu menjadi bersaudara. (Ingatlah pula ketika itu) kamu berada di tepi jurang neraka, lalu Allah menyelamatkan kamu dari sana. Demikianlah Allah menerangkan ayat-ayat-Nya kepadamu agar kamu mendapat petunjuk." *(Q.S Ali Imran:103)*

Dijelaskan dalam tafsir ringkas Kemenag mengenai QS *Ali Imran* 103, Pada ayat ini, Allah memerintahkan umat Muslim untuk menjaga persatuan dan kesatuan. Mereka diingatkan untuk berpegang teguh pada ajaran agama Allah dan saling membantu dalam mempererat tali persaudaraan agar tidak terjatuh dari jalan-Nya. Selain itu, Allah melarang perpecahan, permusuhan, dan rasa dengki, karena hal-hal tersebut dapat melemahkan umat dan membuat mereka mudah dihancurkan. Demikianlah, Allah senantiasa menjelaskan ayat-ayat-Nya kepadamu agar kamu senantiasa mendapatkan petunjuk dan tetap bersatu dalam persaudaraan serta kekeluargaan. (Kemenag, 2022).

Ayat ini mengajarkan bahwa dengan bersatu dan tidak berpisah-pisahan, umat dapat mencapai kesatuan dalam menjalankan agama. Pada algoritma *Random Forest*, pendekatan ini menggambarkan bagaimana *ensemble learning* (penggabungan hasil dari beberapa model *decision tree*) dalam *Random Forest* bisa meningkatkan akurasi dan performa dalam mendeteksi serangan MITM (Z. Qu et al., 2023). Dengan menggunakan variasi model *decision tree* yang berbeda, *Random Forest* dapat mengatasi kelemahan dari model individu dan menghasilkan prediksi yang lebih baik. Ayat ini juga menyebutkan bahwa Allah memberikan petunjuk kepada umat-Nya (Kemenag, 2022). Begitu juga algoritma *Random Forest* memberikan "petunjuk" atau indikator dalam bentuk hasil

klasifikasi yang membantu mendeteksi serangan. Ini mirip dengan cara Allah memberikan petunjuk kepada umat-Nya untuk menghindari keburukan dan mencapai keselamatan.

Pemilihan *Random Forest* didasarkan pada keunggulannya dalam mengatasi masalah *overfitting* dan popularitasnya dalam deteksi serangan (Ekawijana et al., 2024). Dalam penelitiannya mereka mengidentifikasi serangan DDoS dengan tingkat akurasi mencapai 98%. Pada penelitian lain, metode *Random Forest* digunakan untuk mendeteksi serangan *Man in the Middle* pada sistem *Supervisory Control and Data Acquisition* menghasilkan akurasi sebesar 99.93% (Septiano, 2021). *Random Forest* cocok digunakan dalam klasifikasi biner di mana terdapat proses *bagging* yang dilakukan pada pohon yang dibuat sehingga dapat mengatasi *overfitting* (Durairaj et al., 2022). Deteksi serangan MITM sangat penting dalam menjaga layanan dan keamanan data dalam jaringan *Smart Grid* yang rentan terhadap serangan.

1.2 Rumusan Masalah

Berdasarkan latar belakang, rumusan masalah pada penelitian ini adalah bagaimana performa algoritma *Random Forest* dalam mendeteksi serangan *Man in The Middle* pada protokol Modbus TCP/IP di jaringan *Smart Grid.*

1.3 Batasan Masalah

Adapun untuk batasan masalahnya yaitu :

1. Hanya melakukan deteksi untuk serangan *Man in The Middle*.
2. Data yang digunakan berasal dari zenodo.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini ialah untuk mengetahui performa algoritma *Random Forest* dalam mendeteksi serangan *Man in The Middle* pada protokol Modbus TCP/IP di jaringan *Smart Grid.*

1.5 Manfaat Penelitian

Manfaat dari penelitian ini ialah:

1. Temuan dan solusi dalam penelitian ini diharapkan dapat membantu meningkatkan keamanan jaringan IoT pada *Smart Grid*, khususnya pada protokol Modbus TCP/IP dari serangan *Man in The Middle*, yang dapat membantu industri dan penyedia layanan IoT mengembangkan sistem keamanan yang lebih kuat dan efektif.
2. Dapat dijadikan sebagai acuan untuk penelitian selanjutnya dalam mengembangkan sistem keamanan untuk mencegah serangan *Man in The Middle* pada jaringan *Smart Grid.*

BAB II

STUDI PUSTAKA

2.1 Penelitian Terkait

Penelitian yang dilakukan oleh Elrawy dan temannya pada tahun 2023 melakukan deteksi dan klasifikasi serangan pada jaringan *Smart Grid* menggunakan metode *Support Vector Machine* (SVM) dan *Principal Component Analysis* (PCA). Penelitian ini mengusulkan pendekatan *Hybrid Network Intrusion Detection System* yang menggabungkan deteksi anomali berbasis pembelajaran mesin dan berbasis tanda tangan untuk mendeteksi dan mengklasifikasikan serangan. PCA digunakan untuk deteksi anomali sedangkan *Random Forest* digunakan untuk klasifikasi serangan. Pendekatan ini mampu mengidentifikasi serangan MITM dengan menggunakan *Random Forest* dan menunjukkan tingkat akurasi yang tinggi dalam deteksi serangan, dari 97.6% hingga 100%, dengan rata-rata skor *f1 score* di atas 98% (Elrawy et al., 2023).

Penelitian lainnya dilakukan oleh Ghosh dan temannya pada tahun 2023, mengenai deteksi serangan DoS dan *Man in the Middle* pada protokol Modbus di sistem kontrol industri menggunakan *Support Vector Machine* dan *Random Forest.* Penelitian ini menggunakan analisis entropi berbasis klasifikasi untuk mendeteksi anomali pada lalu lintas sistem kontrol industri, khususnya protokol Modbus TCP/IP. Dengan menggunakan dua fitur Modbus tambahan dan tiga fitur TCP/IP, serangan DoS dapat diklasifikasikan dengan akurasi rata-rata 97.87%, sedangkan serangan *Man in the Middle* dapat diklasifikasikan dengan akurasi

86.41% menggunakan *Random Forest*. *Random Forest* terbukti paling efektif dalam mendeteksi serangan *Man in the Middle* pada lalu lintas sistem kontrol industri (T. Ghosh et al., 2023).

Penelitian lainnya mengenai deteksi intrusi pada protokol Modbus di jaringan *Supervisory Control And Data Acquisition* (SCADA) dilakukan oleh Haicheng Qu pada tahun 2018. Pada penelitian ini menggunakan berbagai algoritma *machine learning* seperti *Decision Tree, Random Forest, K-Nearest Neighbors,* SVM, dan *One-Class Support Vector Machine*. Penelitian ini menggunakan dataset nyata pipa gas dan dataset tangki penyimpanan. Kumpulan data mencakup lalu lintas jaringan, kontrol proses, dan fungsi pengukuran proses dari serangkaian 28 serangan pada dua sistem kontrol industri menggunakan protokol lapisan aplikasi Modbus. Hasilnya ialah metode *One-Class Support Vector Machine* (OCSVM) memiliki akurasi tinggi sebesar 98,04% (H. Qu et al., 2018).

Pada penelitian lainnya yang dilakukan oleh Septiano pada tahun 2021, meneliti tentang deteksi serangan MITM pada jaringan *Supervisory Control And Data Acquisition* (SCADA) pada protokol IEC 104 menggunakan *Random Forest*. Pada SCADA terdapat protokol IEC 104 yang rentan terhadap MITM karena pengiriman data dilakukan tanpa adanya enkripsi. Penelitian ini menggunakan *Snort* IDS untuk mendeteksi paket yang ada. Kemudian dilakukan klasifikasi menggunakan *Random Forest* untuk membedakan paket serangan dan paket normal. Pada penelitian ini *Random Forest* menghasilkan akurasi sebesar 99,93%

dengan nilai OBB *Score* 0,07% dan *Un-Detection* rate sebesar 0,05% (Septiano, 2021).

Penelitian mengenai serangan *Man in The Middle* pada jaringan SCADA juga dilakukan oleh Farhan pada tahun 2021 dengan menggunakan metode *Decission Tree. P*enelitian ini melakukan deteksi serangan MITM pada protokol IEC 104. Data yang digunakan pada penelitian ini tidak seimbang dengan jumlah kelas normal lebih banyak daripada kelas serangan. Ketidakseimbangan data ini akan menyebabkan data mayoritas lebih mudah dikenali daripada data minoritas. Oleh karena itu, pada penelitian ini digunakan algoritma *Synthetic Minority Oversampling Technique* (SMOTE) untuk mengatasi ketidakseimbangan kelas. Hasil terbaik dari *decision tree* menggunakan *oversampling* dengan pembagian data *training* sebesar 60% dan *testing* sebesar 40%. Hasilnya, diperoleh *True Positive Rate* (TPR) sebesar 95.31%, *False Positive Rate* (FPR) sebesar 0.01%, *True Negative Rate* (TNR) sebesar 99.98%, *False Negative Rate* (FNR) sebesar 4.68%, dan tingkat akurasi sebesar 97.52% (Farhan, 2021).

Penelitian lainnya mengenai *Intrussion Detection System* (IDS) dilakukan oleh Khan dan temannya pada tahun 2021. Penelitian ini mengusulkan sistem IDS berbasis fitur untuk jaringan *Smart Grid.* Pada penelitian ini mereka meneliti tentang deteksi serangan pada *Smart Grid* menggunakan dataset KDD99 dan NSL KDD dimana terdapat lebih dari 20 jenis serangan menggunakan beberapa algoritma *machine learning.* Hasil yang diperoleh menunjukkan bahwa klasifikasi menggunakan *Random Forest* dan *Neural Network* telah mengungguli klasifikasi lainnya. Penelitian ini berhasil mencapai tingkat alarm palsu sebesar 0,5% pada

dataset KDD99 dan 0,08% pada dataset NSLKDD. Tingkat deteksi dan akurasi pengujian rata-rata mencapai 99% untuk kedua dataset (Khan et al., 2021).

Tabel 2.1 Penelitian terkait

No.	Nama Peneliti	Objek	Metode yang digunakan	Perbedaan Penelitian
1.	(Elrawy et al., 2023)	Deteksi & klasifikasi MITM pada jaringan *Smart Grid*	PCA & SVM	Pendekatan HNIDS menggunakan algoritma SVM dan PCA efektif mendeteksi (MITM) pada *Smart Grid* dengan akurasi 97.6% hingga 100%, dan rata-rata skor F1 di atas 98%
2.	(T. Ghosh et al., 2023)	Deteksi serangan Dos & MITM pada protokol Modbus di sistem kontrol industri	*Random Forest*, *BayesNet*, SVM, *Naïve Bayes*, MLP, J48	Penggunaan metode *Random Forest* dalam mendeteksi serangan *Man In The Middle* (MITM) pada jaringan SCADA menghasilkan tingkat akurasi sebesar 86.41%.
3.	(H. Qu et al., 2018)	Deteksi intrusi pada protokol Modbus di SCADA menggunakan *machine learning*	*Decision Tree*, *KNN*, SVM, dan *One-Class Support Vector Machine*.	Metode *One-Class Support Vector Machine* (OCSVM) memiliki akurasi tinggi sebesar 98,04%.
4.	(Septiano, 2021)	Deteksi serangan MITM pada protokol IEC 104 pada jaringan SCADA	*Random Forest*	Menggunakna Snort IDS untuk mendeteksi serangan kemudian diklasifikasikan menggunaan *Random Forest* menghasilkan akurasi 99,93%
5.	(Farhan, 2021)	Deteksi serangan MITM pada protokol IEC 104 pada jaringan SCADA	*Decission Tree*	Penggunaan *Decision Tree* dalam mendeteksi serangan *Man In The Middle* (MITM) pada jaringan SCADA menghasilkan tingkat akurasi sebesar 97.52%.

No.	Nama Peneliti	Objek	Metode yang digunakan	Perbedaan Penelitian
6.	(Khan et al., 2021)	Deteksi serangan pada jaringan *Smart Grid* menggunakan dataset KDD99 dan NSLKDD	*K-Nearest Neighbor, Neural Network, Decission Tree, Random Forest*	*Random Forest* dan *Neural Network,* berkinerja lebih baik dari metode lainnya dalam akurasi untuk kedua dataset KDD99 dan NSLKDD.

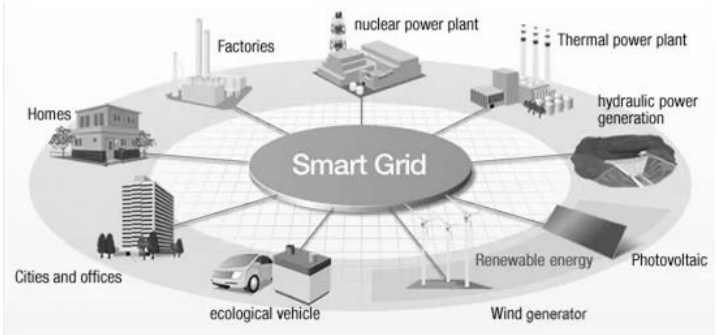
Pada tabel 2.1 terdapat beberapa penelitian mengenai deteksi serangan menggunakan berbagai algoritma seperti *Support Vector Machine, Random Forest, Decission Tree, K-Nearest Neighbor,* dan *Neural Network.* Dari beberapa algoritma pada tabel 2.1, beberapa algoritma memiliki akurasi yang tinggi dalam mendeteksi serangan, seperti algoritma *Random Forest* memiliki akurasi sebesar 99,93% pada penelitian (Septiano, 2021) dan akurasi sebesar 86,41% pada penelitian (T. Ghosh et al., 2023), algoritma SVM dengan akurasi 97,6% pada penelitian (Elrawy et al., 2023), dan juga algoritma *Decision Tree* dengan akurasi sebesar 97,52% pada penelitian (Setiadi, 2021).

Kebaruan pada penelitian ini akan memfokuskan pada pengembangan model deteksi serangan MITM pada protokol Modbus TCP/IP menggunakan algoritma *Random Forest*. Penelitian ini menggunakan data operasional berdasarkan hasil eksperimen laboratorium menggunakan protokol komunikasi Modbus TCP/IP pada jaringan *Smart Grid* yang diambil dari data publik zenodo untuk melatih model, sehingga model deteksi dapat mempelajari pola komunikasi dalam jaringan dan mendeteksi anomali yang mengindikasikan serangan MITM dengan tingkat akurasi yang tinggi. Metode *Random Forest* dipilih karena

kemampuanya yang lebih tahan dari *overfitting* dan juga dari beberapa penelitian serupa tentang deteksi serangan, algoritma ini memiliki nilai akurasi yang tinggi. Sehingga, diharapkan penelitian ini dapat memberikan sumbangan yang berarti dalam meningkatkan keamanan pada jaringan *Smart Grid*.

2.2 *Smart Grid*

Smart Grid ialah konsep jaringan listrik pintar yang mengintegrasikan berbagai entitas seperti pembangkit listrik, pabrik, konsumen, dan perusahaan melalui komunikasi dua arah (Khan et al., 2021). *Smart Grid* adalah jaringan listrik yang memanfaatkan teknologi informasi dan komunikasi untuk mengumpulkan data secara *real-time* dan mengoptimalkan pengiriman listrik dari sumber daya yang berbeda ke konsumen (Sinaga et al., 2021). *Smart Grid* menghubungkan antara perangkat keras, perangkat lunak, dan infrastruktur cerdas untuk memungkinkan manajemen energi yang efisien, pemantauan, dan respon terhadap jaringan listrik (Durairaj et al., 2022). Hal ini dirancang untuk meningkatkan efisiensi, keandalan, keamanan, dan kemajuan teknologi pada jaringan listrik.



Gambar 2.1 Ilustrasi *Smart Grid*
Sumber : medium.com

Pada gambar 2.1 terlihat bahwa *Smart Grid* mengintegrasikan berbagai macam entitas diantaranya terdapat produsen seperti berbagai macam pembangkit listrik dan konsumen misalnya kendaraan listrik, pabrik, rumah juga perkantoran. Teknologi *Smart Grid* memungkinkan integrasi yang lebih baik antara sumber energi terbarukan, seperti pembangkit listrik tenaga surya, air maupun angin. Dengan kemampuan monitoring dan kontrol yang lebih baik, *Smart Grid* dapat mendeteksi dan merespons gangguan jaringan dengan lebih cepat, mengurangi waktu pemadaman listrik, serta memberikan informasi yang lebih baik kepada konsumen tentang penggunaan energi (A. Ghosh & Kole, 2021). Dimana hal ini akan sangat memudahkan tata kelola jaringan listrik, *Smart Grid* bertujuan untuk mengoptimalkan sistem pengiriman listrik, meningkatkan keandalan, dan mengatasi masalah lingkungan dengan memanfaatkan teknologi informasi untuk meningkatkan konektivitas, efisiensi, dan keamanan (Sinaga et al., 2021).

2.3 Protokol *ModBus* TCP/IP

Berbagai protokol komunikasi digunakan pada *Smart Grid.* Karena skalanya begitu besar dan berfungsi untuk menghubungkan beberapa entitas yang berbeda, maka ada banyak protokol komunikasi pada *Smart Grid* (Elrawy et al., 2023). Salah satunya yaitu Modbus TCP/IP. Modbus (*Modicon Bus*) ialah protokol komunikasi industri yang banyak digunakan di berbagai industri elektronik, termasuk pada *Smart Grid.* Modbus TCP/IP menggunakan protokol TCP/IP yang umum digunakan dalam jaringan komputer, memungkinkan komunikasi antara perangkat dapat terhubung dengan baik (T. Ghosh et al., 2023). Modbus TCP/IP memiliki peran yang penting dalam *Smart Grid* karena berfungsi

untuk mengintegrasikan berbagai perangkat dan sistem didalamnya. Perangkat seperti *smart meters, inverter,* dan *relay* menggunakan protokol Modbus TCP/IP untuk bertukar informasi dengan sistem manajemen energi (Elbez et al., 2018). Melalui protokol ini, data pengukuran, pengendalian, dan monitoring dapat dikirim dan diterima dengan efisien dalam jaringan *Smart Grid.*

Namun, protokol Modbus memiliki kerentanan, yaitu tidak menyediakan enkripsi untuk data yang ditransmisikan, sehingga data mudah diintersepsi dan dibaca oleh pihak ketiga. Modbus juga tidak memiliki mekanisme verifikasi integritas pesan, sehingga pesan yang sedang ditransmisikan dapat diubah oleh penyerang tanpa terdeteksi (T. Ghosh et al., 2023). Protokol ini rentan terhadap serangan *replay*, di mana penyerang dapat menangkap dan mengirim ulang pesan yang valid untuk menyebabkan tindakan yang tidak diinginkan (Elrawy et al., 2023). Sehingga membuatnya menjadi rentan terhadap berbagai serangan khususnya serangan *Man in The Middle* karena penyerang dapat memantau dan memodifikasi komunikasi antara dua perangkat tanpa terdeteksi.

2.4 *Man in The Middle*

Kemudahan yang ditawarkan oleh *Smart Grid* tentunya memiliki kelemahan, salah satunya yaitu data yang dikirimkan tidak dienkripsi, sehinga menjadikanya celah keamanan (Prima, 2022). Banyak serangan yang bisa dilakukan terhadap celah kemanan ini, salah satunya yaitu *Man in The Middle* (MITM). Serangan MITM ialah serangkaian serangan di mana seorang penyerang mengganggu komunikasi antara pengirim dan penerima dengan memecah saluran komunikasi menjadi dua saluran (Setiadi, 2021).

Serangan MITM pada protokol Modbus dapat terjadi dengan penggunaan teknik *ARP poisoning*. Dalam *ARP poisoning*, penyerang memodifikasi tabel *ARP* pengirim untuk mengambil alih saluran komunikasi (Prakasa, 2020). Penyerang dapat menargetkan pengontrol untuk mengambil alih lalu lintas yang datang dari pengontrol, *router* untuk mengambil alih lalu lintas yang datang dari perangkat daya, atau kedua sisi untuk mengambil alih lalu lintas masuk dan keluar (Elrawy et al., 2023). Dalam *port stealing*, penyerang dapat memanipulasi tabel alamat MAC *switch* untuk mengambil alih saluran komunikasi (Septiano, 2021). Hal ini sangat berbahaya karena dapat menimbulkan, penyadapan data, kerugian bahkan kerusakan yang besar pada jaringan *Smart Grid.*

2.5 Model Klasifikasi

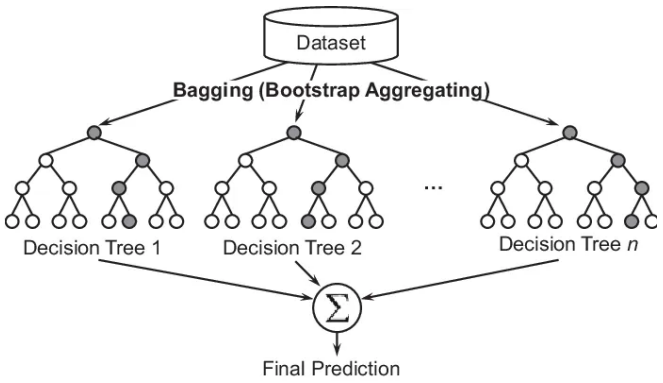
Model klasifikasi merupakan algoritma dalam *machine learning* yang digunakan untuk memprediksi kategori atau label dari suatu data berdasarkan fitur-fitur yang dimilikinya (Prima, 2022). Model ini dilatih menggunakan data yang telah dilabeli untuk mengidentifikasi data *training* ke dalam kelas-kelas atau kategori yang telah ditentukan sebelumnya (Rafsanjani et al., 2022). Deteksi dari model klasifikasi ialah kemampuan model untuk mengidentifikasi pola atau perilaku yang mencurigakan dalam data berdasarkan fiturnya, sehingga dapat digunakan untuk mendeteksi serangan atau ancaman keamanan, seperti serangan DDoS, serangan *malware*, *Botnet* bahkan MITM maupun serangan lainnya (Ekawijana et al., 2024). Model klasifikasi dapat membantu meningkatkan keamanan jaringan dengan mengidentifikasi dan merespons ancaman secara cepat dan efektif.

2.6 *Random Forest*

Random Forest merupakan salah satu algoritma *machine learning* yang biasa digunakan untuk klasifikasi maupun regresi (Kurniabudi et al., 2020). Algoritma ini merupakan gabungan dari algoritma *Decision Tree* dengan teknik *Boostrap Aggregating* (*Bagging*) (Uzlah & Saputra, 2024). *Random Forest* mampu mengatasi masalah *overfitting* pada algoritma *Decision Tree* dengan cara melakukan *bagging* (Durairaj et al., 2022). Pada algoritma ini dibuat beberapa pohon keputusan dari sampel acak yang diambil, kemudian hasil dari tiap pohon akan digunakan untuk memperbaiki akurasi model (Durairaj et al., 2022).

Algoritma ini memiliki beberapa keunggulan yang membuatnya sangat efektif dalam analisis data. *Random Forest* mampu menganalisis data berukuran besar dengan waktu yang singkat. Selain itu, juga menghasilkan tingkat akurasi yang tinggi, yang sangat penting dalam analisis data. *Random Forest* mampu memproses data kontinu dan kategoris dengan baik, sehingga dapat digunakan dalam berbagai jenis dataset (Rafsanjani et al., 2022).

Alur kerja *Random Forest* secara umum dijelaskan pada gambar 2.2.



Gambar 2.2 Ilustrasi *Random Forest*
Sumber : medium.com

Pada gambar 2.2 *Random Forest* membentuk beberapa pohon keputusan dari sebuah dataset yang menghasilkan nilai yang beragam. Pohon ini dihasilkan dari data yang diambil dari proses *bagging*, dimana setiap pohon keputusan dilatih menggunakan sampel data acak yang diambil dari dataset asli dengan teknik *bagging* (Z. Qu et al., 2023). Sehingga setiap pohon keputusan akan menggunakan data yang berbeda-beda. Untuk mendapatkan hasil akhir, *Random Forest* akan melakukan mayoritas *voting* dari seluruh jumlah pohon, sehingga hasilnya tidak mudah *overfitting*.

Langkah-langkah dalam proses klasifikasi dengan menggunakan *Random Forest* menurut (Sholihah & Hermawan, 2023) ialah :

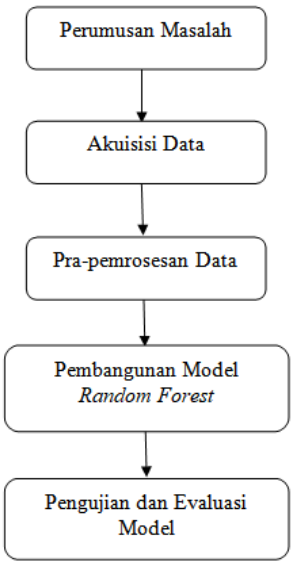
1. Ambil sampel acak berukuran n dari data *training* dengan metode pengambilan dengan pengembalian (*bootstrap sampling*).
2. Bangun pohon dengan data *bootstrap* hingga ukuran maksimum, pilih acak m variabel penjelas dimana (m < p) pada setiap pemisahan, dan lakukan pemisahan terbaik.
3. Ulangi langkah 1 dan 2 hingga k kali untuk menghasilkan k pohon acak.
4. Prediksi hasil respon dari suatu percobaan dengan menggabungkan prediksi dari k pohon, kemudian tentukan hasil prediksi akhir dengan memilih suara terbanyak menggunakan *majority voting*.

BAB III

DESAIN DAN IMPLEMENTASI

3.1 Analisis dan Perancangan

Alur pada penelitian ini akan dipaparkan pada gambar 3.1 untuk memberikan gambaran penelitian secara umum.



Gambar 3.1 Analisis dan Perancangan

3.2 Perumusan Masalah

Pada gambar 3.1 terdapat bagian perumusan masalah dimana pada bagian ini peneliti merumuskan permasalahan mengenai pengembangan model deteksi serangan *Man In The Middle* (MITM) pada protokol Modbus TCP/IP dalam jaringan *Smart Grid.* Karena jaringanya yang kompleks, maka *Smart Grid* memiliki banyak protokol, salah satunya adalah Modbus. Dimana pada protokol Modbus ini, data dikirimkan dengan cepat namun tanpa proses enkripsi, sehingga rawan terhadap serangan, khususnya MITM. Permasalahan utama yang ingin diselesaikan adalah bagaimana mengidentifikasi serangan MITM dengan tingkat

akurasi yang tinggi sehingga dapat meningkatkan keamanan pada jaringan *Smart Grid.*

3.3 Akuisisi Data

Pada tahap ini peneliti mengumpulkan data dan informasi mengenai serangan MITM pada *Smart Grid.* Data ini dihasilkan dari eksperimen laboratorium menggunakan protokol Modbus TCP/IP pada *Smart Grid.* Data ini memiliki 5583 baris data dengan 6 fitur dan 1 label. Seperti yang terlihat pada tabel 3.1 terdapat fitur *Inter Response Time* (IRTT) dalam detik, *Time to Open Connection* (TTOC) dalam detik, *Minimal Time Response* (MITR) dalam detik, *Maximal Time Response Time* (MATR) dalam detik, *Number of Requested Open Connection* (NROC), *ARP poisoning indicator*, dan Label. Data ini memiliki 4 kelas yaitu MITM *two-way attack* (MITM TW) dengan 738 data, MITM *attacking controller* (MITM AC) dengan 670 data, MITM *attacking router* (MITM AR) dengan 737 data dan kelas Normal dengan 3438 data.

Tabel 3.1 Fitur pada dataset yang digunakan

Fitur	Keterangan
Inter Response Time for Traffic (IRTT)	Waktu yang dibutuhkan untuk membangun satu koneksi antara perangkat pengguna dan server.
Time To Open Connection (TTOC)	Waktu yang diperlukan untuk menerima semua respon dari server setelah satu koneksi dibangun.
Minimal Time Response (MITR)	Waktu minimum antara permintaan data yang dikirim dalam satu koneksi.
Maximal Time Response (MATR)	Waktu maksimum antara permintaan data yang dikirim dalam satu koneksi.
Number of Requested Open Connection (NROC)	Jumlah permintaan data yang dikirim dalam satu koneksi antara perangkat pengguna dan server.
ARP poisoning indicator	Indikator serangan *ARP Poisoning.*

3.4 Pra Pemrosesan Data

Pada tahap ini peneliti melakukan pra pemrosesan data untuk mempersiapkan data yang akan digunakan untuk pelatihan model. Pra pemrosesan data akan mencakup pemisahan fitur (*missing value*), pelabelan kelas, pemilihan fitur, serta pembagian data menjadi data latih *(training)* dan data uji *(testing)*. Pembagian data yang digunakan untuk *training* dan *testing* pada penelitian ini dilakukan secara acak (*random sampling*).

3.4.1 *Missing Value*

Pada pra pemrosesan data biasanya ada data yang tidak seimbang atau diluar jangkauan, maka perlu dilakukan adanya perbaikan data untuk menemukan dan memperbaikai data yang tidak akurat ataupun terdapat nilai yang hilang. Pada tahap ini data yang tidak digunakan akan dihilangkan. Pada data ini fitur yang dihapus ialah NROC & *ARP Poisoning*. NROC dihapus karena semua nilainya sama yaitu 13, sehingga tidak mempengaruhi hasil akhirnya. Sedangkan, *ARP Poisoning indicator* akan dihapus karena nilainya hanya 0 dan 1, jika 0 berarti normal dan 1 berarti serangan dan hanya sedikit berpengaruh pada hasil akhir.

3.4.2 Pelabelan Kelas

Pada data ini terdapat label dengan nama kolom Label: 1(Normal), 1000 (MITM *attacking controller*), 2000 (MITM *attacking router*), 500 (MITM *two-way attack*) dengan nilainya 1, 500, 1000 dan 2000. Untuk memudahkan dalam proses pemodelan dan pengujian maka setiap nilai pada label akan di *refactor* menjadi kategorinya masing-masing. Dengan aturan jika 1 adalah kelas normal,

500 adalah kelas MITM *two-way attack*, 1000 adalah MITM *attacking controller* dan 2000 adalah MITM *attacking router*. Pada gambar 3.2 ditunjukkan data sebelum dilakukan pra pemrosesan data. Terlihat terdapat 6 fitur dan 1 label disana. Kemudian pada gambar 3.3 ditunjukan dataset yang telah di lakukan pra pemrosesan data. Dimana terdapat 4 fitur dan 1 label.

Data sebelum dan sesudah dilakukan *missing value* & pelabelan kelas

IRTT	TTOC	MITR	MATR	NROC	ARP poiso	Label: 1(Normal), 1000 (MITM attacking controller), 2000 (MITM attacking router), 500 (MITM two-way attack)
0.004345	0.24373	0.010792	0.048848	13	0	1
0.005715	0.164952	0.009401	0.01604	13	0	1
0.004437	0.161428	0.009643	0.014439	13	0	1
0.0053	0.166849	0.009291	0.021015	13	0	1
0.00436	0.155357	0.009209	0.01589	13	0	1
0.004624	0.186185	0.009429	0.032564	13	0	1
0.00534	0.16396	0.009383	0.020141	13	0	1

Gambar 3.2 Data sebelum dilakukan *missing value* & pelabelan kelas

IRTT	TTOC	MITR	MATR	Label
0.004345	0.24373	0.010792	0.048848	Normal
0.005715	0.164952	0.009401	0.01604	Normal
0.004437	0.161428	0.009643	0.014439	Normal
0.0053	0.166849	0.009291	0.021015	Normal
0.00436	0.155357	0.009209	0.01589	Normal
0.004624	0.186185	0.009429	0.032564	Normal
0.00534	0.16396	0.009383	0.020141	Normal

Gambar 3.3 Data sesudah dilakukan *missing value* & pelabelan kelas

3.4.3 Split Data

Pada tahap ini data dibagi menjadi 2 yaitu data *testing* dan data *training.* Data latih *(training)* digunakan untuk membangun model *Random Forest*. Sedangkan data uji *(testing)* berfungsi untuk mengevaluasi performa model. Dalam penisahan data, penulis menggunakan *stratify* yang berfungsi untuk memastikan distribusi kelas pada data *training* & *testing* tetap seimbang, sesuai dengan distribusi kelas yang ada pada dataset (Ye et al., 2013). Untuk pembagian

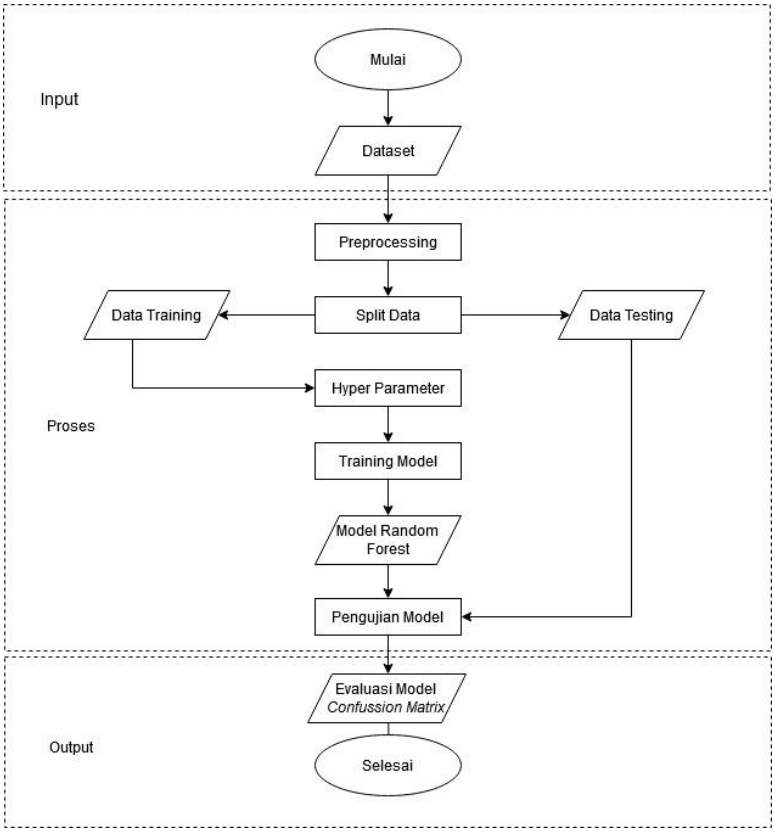
data pada penelitian ini dibagi menjadi 5 model dengan pembagian data dijelaskan pada tabel 3.2 di bawah ini.

Tabel 3.2 Pembagian model

No.	**Model**	***Training***	***Testing***
1	Model 1	50%	50%
2	Model 2	60%	40%
3	Model 3	70%	30%
4	Model 4	80%	20%
5	Model 5	90%	10%

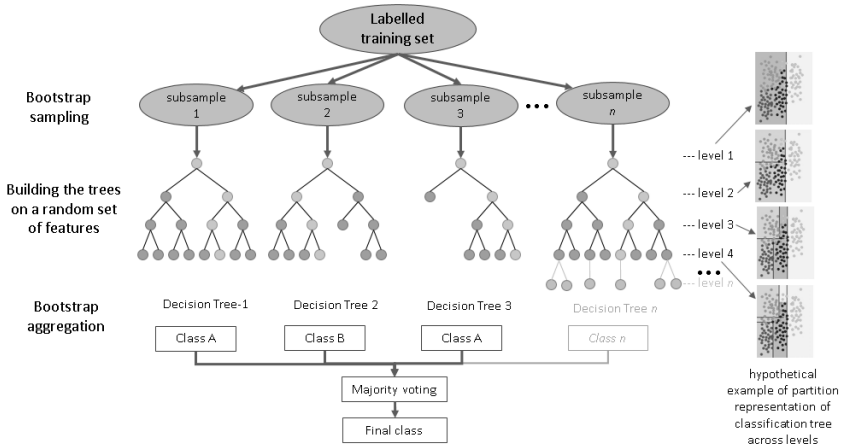
Pada tabel 3.2 diatas penelitian ini akan dibuat 5 model dengan melakukan variasi pada pembagian data *testing*, *training* dan juga *hyperparameter*. *Hyperparameter* yang digunakan ialah *n_estimator* yaitu jumlah pohon yang akan dibuat pada model dengan nilai 100 dan 200 serta *max_features* yaitu jumlah fitur yang akan digunakan pada setiap pohon yang dihitung dari *sqrt* (akar) atau *log2* dari jumlah fitur pada dataset. Dengan adanya pembagian ini dapat memberikan kinerja yang lebih baik untuk evaluasi kelima model sehingga dapat mencegah *overfitting* pada model. Dari kelima model ini akan diambil model mana dengan akurasi terbaik untuk mengidentifikasi serangan *Man in The Middle* pada jaringan *Smart Grid.*

3.5 Pembangunan Model *Random Forest*



Gambar 3.4 Alur data algoritma *Random Forest* pada penelitian ini

Gambar 3.4 menunjukan alur data bagaimana *Random Forest* melakukan deteksi. Pada tahap pertama ialah menyiapkan dataset, kemudian pada tahap proses ada beberapa tahapan. Pada tahap *preprocessing*, dilakukan pra pemrosesan data seperti yang dijelaskan sebelumnya pada poin 3.4, kemudian data dibagi menjadi 2 yaitu data latih *(training)*, dan data uji *(testing)* dengan proporsi sesuai dataset menggunakan *stratify*. Kemudian pada data *training* dilakukan pengaturan *hyperparameter* dengan skenario dan komposisi *train* & *test* seperti yang diuraikan pada sub bab 3.4. Setelah itu dilakukan pelatihan pada model menggunakan *Random Forest*.



Gambar 3.5 Model *Random Forest*
Sumber : catalyst.earth

Pada gambar 3.5 ditunjukan bahwa model *Random Forest* akan membuat beberapa pohon keputusan untuk melakukan prediksi. Setiap pohon dibangun menggunakan data *subsample* yang berbeda antara tiap pohonnya. Pada tahap pelatihan (*training model*) peneliti melakukan *boostrap sampling* untuk mengambil *subsample* secara acak dari data *training.* Dari proses *boostrap sampling*, dilakukan pemilihan *threshold* terbaik dari-data yang ada untuk membuat pohon. Disini peneliti menggunakan *Gini Impurity* untuk menentukan *threshold* terbaik, dimana yang nilai *Gini Impurity* paling mendekati 0 adalah yang terbaik. Berikut ini adalah rumus perhitungannya yang ditunjukkan pada persamaan 3.1 (Uzlah & Saputra, 2024).

$$Gini = 1 - \sum_{i=1}^{C} (p_i)^2 \tag{3.1}$$

Keterangan:

c = Jumlah kelas dalam data.

p_i = proporsi kelas i dalam data.

$\Sigma (p_i)^2$ = jumlah kuadrat dari proporsi setiap kelas dalam data.

Pada gambar 3.6 dibawah ini terdapat sampel dataset yang telah diambil dengan teknik *boostrap*.

IRTT	TTOC	MITR	MATR	Label
0.004211	0.176778	0.009152	0.037134	Normal
0.005092	0.279026	0.009620	0.049375	Normal
0.005205	0.221798	0.009568	0.044280	Normal
0.021599	0.433112	0.015182	0.048865	MITM attacking controller
0.012654	0.472248	0.016620	0.047329	MITM attacking router
0.025824	0.535141	0.025212	0.055946	MITM two-way attack

Gambar 3.6 Sampel *datasets*

Dengan menggunakan nilai *Gini Impurity* terendah peneliti mencari *threshold* terbaik. Pertama tentukan nilai tengah dari 2 data yang berdekatan. Untuk fitur IRTT hasilnya ialah seperti terlihat pada gambar 3.7 dibawah ini.

Fitur IRTT
Threshold 1 = (0.004211+0.005092) / 2 = 0.0046515
Threshold 2 = (0.005092+0.005205) / 2 = 0.0051485
Threshold 3 = (0.005205+0.012654) / 2 = 0.0089295
Threshold 4 = (0.021599+0.012654) / 2 = 0.0089295
Threshold 5 = (0.012654+0.025824) / 2 = 0.0192390

Gambar 3.7 Nilai tengah fitur IRTT

Pada gambar 3.7 terdapat 5 *threshold* pada fitur IRTT sehingga perlu dicari nilai *threshold* terbaik dengan menggunakan persamaan 3.1. Karena *threshold* 1 = 0.0046515, maka nilai fitur IRTT yang kurang dari *threshold* 1 akan masuk dalam *subset* 1 yaitu kelas *normal*. Sedangkan yang lebih dari *threshold* 1 akan masuk *subset* 2 yaitu terdapat 2 kelas normal, 1 MITM *attacking controller* (MITM AC), 1 MITM *attacking router* (MITM AR) dan 1 MITM *two-way attack* (MITM TW). Dengan menggunakan persamaan 3.1 dapat diperoleh *threshold* terbaik dengan cara sebagai berikut:

Gini (Subset 1) = $1 - \left(\frac{1}{1}\right)^2 = 0$

Gini (Subset 2) = $1 \ - \left(\frac{2}{5}\right)^2 - \left(\frac{1}{5}\right)^2 - \left(\frac{1}{5}\right)^2 - \left(\frac{1}{5}\right)^2 = 0{,}72$

Gini Total = $\left(\frac{1}{6}\right) * 0 - \left(\frac{5}{6}\right) * 0{,}72 = 0{,}6$

Dari perhitungan diatas didapatkan nilai *Gini Impurity* dari *threshold* 1 ialah 0,6. Kemudian dilakukan perhitungan untuk semua *threshold* sehingga dihasilkan nilai seperti pada gambar 3.8 dibawah ini.

IRTT	Nilai	Gini Impurity
T1	0.0046515	0.6
T2	0.0051485	0.5
T3	**0.0089295**	**0.333**
T4	0.0171265	0.4167
T5	0.0192392	0.4167

Gambar 3.8 Nilai *Gini Impurity* dari tiap *threshold*

Gambar 3.8 menunjukkan bahwa nilai *threshold* terbaik terdapat di *threshold* ke 3 dengan nilai 0.0089295 dengan nilai *Gini Impurity* sebesar 0,333. Langkah ditas diulangi sampai mendapatkan nilai *threshold* terbaik dari tiap fitur yang ada. Setelah melakukan perhitungan maka didapatkan nilai *threshold* terbaik dari tiap fitur, seperti terlihat pada gambar 3.9 dibawah ini.

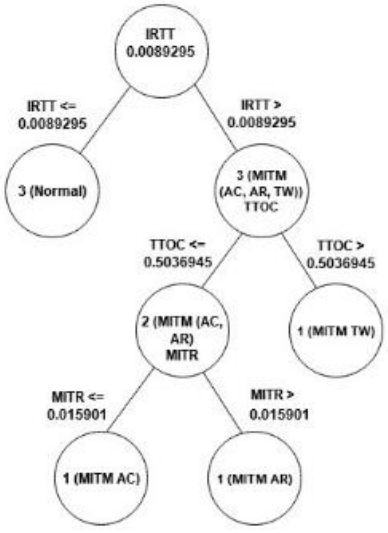
Fitur	Threshold		Gini Impurity
IRTT	**T3**	**0.0089295**	**0.333**
TTOC	T3	0.356069	0.333
MITR	T3	0.012401	0.333
MATR	T2	0.0458045	0.5

Gambar 3.9 Nilai *threshold* terbaik dari tiap fitur

Pada gambar 3.9 terlihat bahwa nilai *Gini Impurity* dari fitur IRTT, TTOC dan MITR sama. Oleh karena itu fitur yang dipilih untuk membuat *node root*

diambil dengan menggunakan pengacakan. Sehingga yang terpilih ialah fitur IRTT sebagai *node root* dengan nilai 0.0089295. Kemudian menentukan nilai yang masuk *leaf* kanan & kiri. Dimana yang nilai fitur IRTT kurang dari *threshold* yang telah ditentukan akan masuk *leaf* kiri. Karena pada gambar 3.6 semua nilai fitur IRTT pada kelas normal kurang dari *threshold* maka semuanya masuk *leaf* kiri. Untuk data lainya (MITM AR, MITM AC dan MITM TW) masuk *leaf* kanan karena nilainya melebihi *threshold*.

Karena pada *leaf* kiri sudah *homogen*, maka pembuatan cabang selesai. Sedangkan pada *leaf* kanan masih *heterogen*, maka pembuatan cabang masih perlu dilakukan. Untuk penentuan cabang perlu dicari fitur dengan *threshold* terbaik dengan cara yang sama seperti sebeulmnya, namun dari data yang tersisa dari keseluruhan sampel data. Kemudian ulangi langkahnya sampai semua *leaf* sudah *homogen*. Untuk hasilnya terlihat pada gambar 3.9 dibawah ini.



Gambar 3.10 Pohon yang dihasilkan dari data *training*

Pada gambar 3.10 menunjukkan pada *node root* terdapat fitur IRTT dengan nilai *threshold* 0.0089295, kemudian *leaf* kiri terdapat data dengan kelas

normal. Sehingga ketika data *testing* masuk *leaf* kiri maka dapat dikategorikan sebagai data dengan kelas normal, begitu juga sebaliknya. Sedangkan untuk bagian *leaf* kanan terdapat kelas serangan, yaitu MITM *attacking controller* (MITM AC), MITM *attacking router* (MITM AR) dan MITM *two-way attack* (MITM TW) dengan *threshold* dan nilainya masing-masing. Setelah model pohon berhasil dibuat, maka data *testing* bisa dimasukkan. Sehingga tiap pohon yang dibangun dapat melakukan prediksi.

Pada algoritma *Random Forest* pembangunan pohon dilakukan sebanyak *n_estimator*. Tiap pohon yang dibangun menggunakan *subsample* yang berbeda-beda sehingga menghasilkan *decision* yang berbeda-beda. Dari hasil ini dilakukan penggabungan (*aggregating*) untuk menghasilkan keputusan yang akurat. Dengan menggunakan *majority voting* akan diambil kelas final dalam menentukan kelas dari data yang ada. Setelah model selesai dilatih, akan dilakukan pengujian menggunakan data *testing* untuk mengevaluasi kinerja model. Model akan diuji untuk melihat seberapa baiknya model dalam mendeteksi serangan MITM pada data baru. Penelitian ini membagi data menjadi 5 model seperti pada tabel 3.2. Pada masing-masing model akan diimplementasikan algoritma *Random Forest* dengan pembagain *training*, *testing* dan *hyperparameter* yang berbeda.

Kemudian kinerja model akan dievaluasi menggunakan *confusion matrix*, seperti akurasi, presisi, *recall*, dan *f1-score*. Akurasi memberikan gambaran umum tentang seberapa baik model dapat memprediksi secara keseluruhan yang ditunjukkan pada persamaan 3.2. Presisi mengukur sejauh mana prediksi positif yang dilakukan oleh model benar, dibandingkan dengan total prediksi positif yang

dihasilkan, seperti dalam persamaan 3.3. *Recall* memberikan informasi tentang seberapa banyak *instance* positif yang diprediksi dengan benar dari semua *instance* yang sebenarnya positif yang ditunjukkan pada persamaan 3.4. Sedangkan, *f1-score* memberikan keseimbangan antara presisi dan *recall*, yang berguna saat ada ketidakseimbangan kelas dalam data yang ditunjukkan pada persamaan 3.5.

1. *Accuracy*

   Akurasi adalah rasio dari jumlah prediksi yang benar (positif dan negatif) dengan jumlah total data (Kurniabudi et al., 2020).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} x100\% \quad (3.2)$$

   TP: *True Positive* (jumlah kasus positif yang diprediksi dengan benar).
   TN: *True Negative* (jumlah kasus negatif yang diprediksi dengan benar).
   FP: *False Positive* (jumlah kasus negatif salah diprediksi sebagai positif.
   FN: *False Negative* (jumlah kasus positif salah diprediksi sebagai negatif).

2. Presisi

   Presisi adalah rasio dari jumlah prediksi positif yang benar dengan jumlah total prediksi positif dan mengukur seberapa akurat model dalam mengidentifikasi kasus positif (Kurniabudi et al., 2020).

$$Precision = \frac{TP}{TP+FP} x100\% \quad (3.3)$$

3. *Recall*

   Recall adalah rasio dari jumlah prediksi positif yang benar dengan jumlah total kasus positif dalam data. *Recall* mengukur seberapa baik model dalam menangkap kasus positif (Uzlah & Saputra, 2024).

$$Recall = \frac{\text{TP}}{\text{TP+FN}} x100\% \tag{3.4}$$

4. F1 *Score*

   F1 *Score* adalah ukuran gabungan dari presisi dan *recall* dan berfungsi untuk memberikan keseimbangan antara presisi dan *recall* (Uzlah & Saputra, 2024).

$$\text{F1 } Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} x100\% \tag{3.5}$$

Semakin tinggi nilai akurasi, presisi, *recall*, dan *F1-Score*, semakin baik pula kinerja model deteksi serangan MITM. Hasil evaluasi akan digunakan untuk menentukan seberapa baiknya model dalam mengidentifikasi serangan MITM.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Uji Coba

Pada bagian ini peneliti melakukan pengujian terhadap model *Random Forest* yang telah dibuat. Pengujian dilakukan dalam beberapa model seperti pada tabel 3.2, dimana pada setiap model terdapat pembagian data latih *(training)* dan data uji *(testing)*. Pada setiap model akan diuji menggunakan *hyperparameter* yang telah ditentukan pada sub bab 3.2. Tujuan dari pengujian ini ialah untuk mengetahui performa metode *Random Forest* dalam melakukan deteksi serangan *Man in The Middle* dengan menggunakan data sebanyak 5583 data. Pengujian dibagi menjadi 3 bagian dengan rincian pada gambar 4.1 di bawah ini.

Uji Coba	Kelas	Jumlah data	Total data
1	Normal + MITM *Two Way Attack* (MITM TW)	3438 + 738	4176
2	Normal + MITM *Attacking Controller* (MITM AC)	3438 + 670	4108
3	Normal + MITM *Attacking Router* (MITM AR)	3448 + 737	4175

Gambar 4.1 Pembagian Pengujian

Pada Gambar 4.1 terlihat setiap uji coba memiliki kombinasi kelas normal dan 1 kelas MITM yang berbeda dengan jumlah data yang berbeda. Pengujian 1 dilakukan pada kelas Normal dan MITM TW kemudian dilakukan *split data* hingga mendapatkan nilai akurasi, dan diulangi untuk pengujian 2 & 3. Masing-masing uji coba dibagi lagi dalam 5 model pengujian dengan proporsi data yang berbeda, model yang telah dibangun dilakukan evaluasi untuk mengetahui performa algoritma *Random Forest* dalam mendeteksi serangan *Man in The Middle* menggunakan *confussion matrix*.

4.2 Hasil Uji Coba

Pada sub bab hasil uji coba ini, peneliti memaparkan hasil dari uji coba skenario yang telah dilakukan menggunakan 5583 data pada jaringan *Smart Grid* yang berasal dari zenodo dataset. Uji coba dilakukan dengan 3 kombinasi seperti pada gambar 4.1. Setiap kombinasi kemudian dilakukan *preprocessing*, *missing value*, pelabelan kelas, *split data* hingga evaluasi model. Penelitian ini dilakukan dengan 5 skenario pengujian yaitu dengan membagi data dengan rasio yang berbeda-beda dengan *hyperparameter* seperti yang telah ditentukan pada sub bab 3.4.3. Masing-masing model dilakukan pengujian menggunakan algoritma *Random Forest* untuk mengukur performanya.

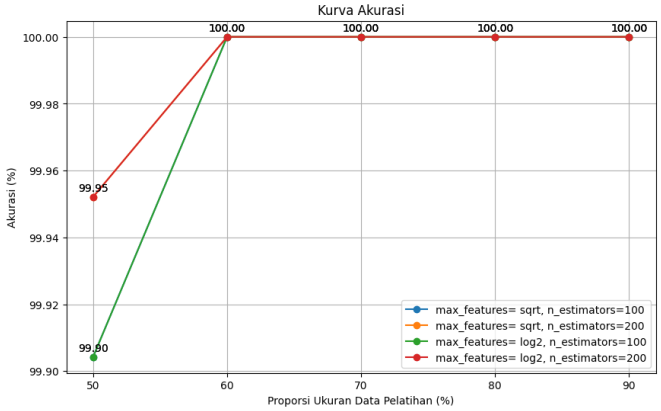
4.1.1 Uji Coba 1

Pada uji coba 1 menggunakan 4176 data dengan 3438 data kelas normal dan 738 data kelas MITM *two-way attack* (TW) dengan 5 pengujian seperti pada gambar 4.2 di bawah ini.

No.	Uji Coba 1	*Training*		*Testing*	
		Proporsi	Jumlah	Proporsi	Jumlah
1	Model 1	50%	2088	50%	2088
2	Model 2	60%	2505	40%	1671
3	Model 3	70%	2923	30%	1253
4	Model 4	80%	3340	20%	836
5	Model 5	90%	3758	10%	418

Gambar 4.2 Pembagian model pada uji coba 1

Terlihat pada gambar 4.2, uji coba ini memiliki 5 pengujian mulai dari model 1 menggunakan proporsi 50% data latih dengan 2088 data dan 50% data uji dengan 2088 data hingga model 5 dengan proporsi 90% data latih sebanyak 3758 data dan 10% data uji sebanyak 418 data. Di bawah ini ialah hasil pengujian dari 5 model.

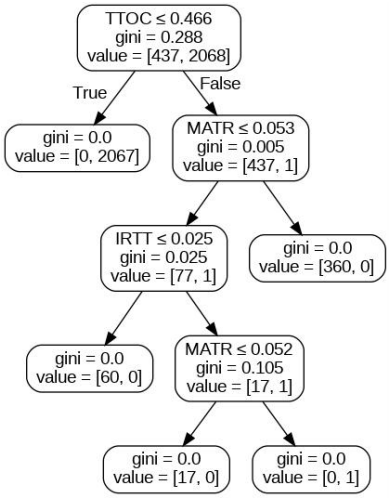


Gambar 4.3 Kurva uji coba 1

Pada gambar 4.3 terlihat bahwa secara umum, hasil menunjukkan bahwa akurasi model meningkat seiring dengan bertambahnya proporsi data pelatihan. Hal ini disebabkan oleh semakin banyaknya data pelatihan yang membuat model mempelajari pola data dengan lebih baik, sehingga menghasilkan prediksi yang lebih akurat. Pada proporsi *training* 50%, rata-rata akurasi mencapai 99.92%. Sedangkan, pada proporsi *training* 60% hingga 90%, semua kombinasi parameter menghasilkan akurasi 100% menunjukkan bahwa model mampu memprediksi semua data uji dengan benar. Hasil akurasi terbaik terdapat pada pengujian kedua dengan menggunakan perbandingan 60% data *training* sebanyak 2505 data dan 40% data *testing* sebanyak 1671 data dengan masing-masing *hyperparameter*. Di bawah ini ialah gambar dari pohon terbaik pada uji coba 1.

```
Pohon terbaik pada (indeks ke-1) memiliki akurasi 1.000000 pada data uji.
Model ini dilatih pada train_test_split = 60% : 40%
dengan kombinasi : n_estimators=100, max_features=sqrt
```

Gambar 4.4 Informasi pohon terbaik



Gambar 4.5 Pohon dengan akurasi terbaik pada uji coba 1

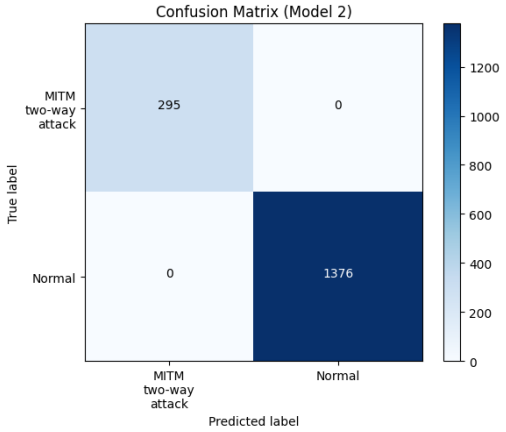
Gambar 4.4 dan 4.5 merupakan informasi dan visualisasi pohon dengan akurasi terbaik pada uji coba 1. Pohon ini terletak pada indeks ke 1 dengan akurasi 100% pada data uji. Pohon ini berasal dari model 2 dengan proporsi 60% data latih dan 40% data uji. Pada *node root threshold* yang digunakan yaitu fitur TTOC dengan nilai 0.466, sehingga yang nilainya < *threshold* akan masuk leaf kiri selebihnya masuk leaf kanan. Dengan nilai *Gini Impurity* 0.288 menunjukan adanya campuran kelas. *Node* ini memiliki proporsi 437 data MITM TW dan 2068 data normal. Pada leaf kiri tidak terdapat *threshold* karena nilai gini = 0, sehingga sudah murni (hanya 1 kelas). Pada node kanan *threshold* yang digunakan ialah MATR dgn nilai 0.053 dengan nilai *Gini Impurity* = 0.005 menunjukkan proporsi data hampir murni. Di bawah ini ialah hasil dari pengujian ini.

Train:Test = 60% : 40%

n_estimators	max_features	train_test_split	accuracy_test
100	sqrt	60% : 40%	1.000000
100	log2	60% : 40%	1.000000
200	sqrt	60% : 40%	1.000000
200	log2	60% : 40%	1.000000

Gambar 4.6 Hasil pengujian model

Hasil klasifikasi pada model pengujian kedua dengan beberapa *hyperparameter* ditunjukkan pada gambar 4.6 dimana semua skenario memiliki nilai akurasi 100% sehingga model terbaiknya diambil pada model pertama dengan kombinasi *n_estimator* = 100, *max_features = sqrt*.




```
Model 2: 60% Train, 40% Test, n_estimators=100, max_features='sqrt'
Accuracy: 1.0
Classification Report:
 Class  Precision    Recall     F1-Score    Support
   0   1.00000000  1.00000000  1.00000000    295.0
   1   1.00000000  1.00000000  1.00000000   1376.0
 avg   1.00000000  1.00000000  1.00000000
True Positive: [295, 1376]
True Negative: [1376, 295]
False Positive: [0, 0]
False Negative: [0, 0]
Training Time: 0.27434802055358887 seconds
Testing Time: 0.023386716842651367 seconds
```

Gambar 4.7 *Confussion matrix* dan *classification report* 60% : 40%

Dari visualisasi dan hasil *classification report* pada gambar 4.7 diketahui nilai dari *precision*, *recall*, dan *f1-score* dari masing-masing kelas. Nilai TP dari dari kelas *MITM two-way attack*, dan kelas Normal ialah 295 dan 1376. Dengan TN adalah 1376 dan 295. FP serta FN adalah 0. Pengujian ini menggunakan waktu *training* sebesar 0.27 detik dan waktu *testing* sebesar 0.02 detik. Dengan menggunakan rumus pada sub bab 3.5 untuk mengukur nilai akurasi, presisi *recall*, dan *f1-score*, Sehingga didapatkan tabel 4.2 dibawah ini.

Tabel 4.1 Hasil *confusion matrix* pengujian model uji coba 1

No.	Kategori	TP	TN	FP	FN	Presisi	*Recall*	*F1-Score*
1.	MITM *two-way attack*	295	1376	0	0	100%	100%	100%
2.	Normal	1376	295	0	0	100%	100%	100%
			Rata - rata			100%	100%	100%

$$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \; x \; 100\% = \frac{295 + 1376}{1671} \; x \; 100\% = 100\%$$

Pada tabel 4.1 didapatkan nilai rata-rata presisi, *recall, f1-score* dan akurasi pada setiap model pengujian. Di bawah ini merupakan nilai akurasi, presisi, *recall,* dan *f1-score* dari uji coba 1.

Tabel 4.2 Hasil performa pada uji coba 1

Model	*Training : Testing*	Akurasi	Presisi	*Recall*	*F1-Score*
1	50% : 50%	99.95%	99.86%	99.97%	99.91%
2	60% : 40%	100%	100%	100%	100%
3	70% : 30%	100%	100%	100%	100%
4	80% : 20%	100%	100%	100%	100%
5	90% : 10%	100%	100%	100%	100%

Terlihat pada tabel 4.2 bahwa akurasi tertinggi terdapat pada model 2 hingga 5 dengan nilai akurasi, presisi, *recall,* dan *f1-score* mencapai 100%.

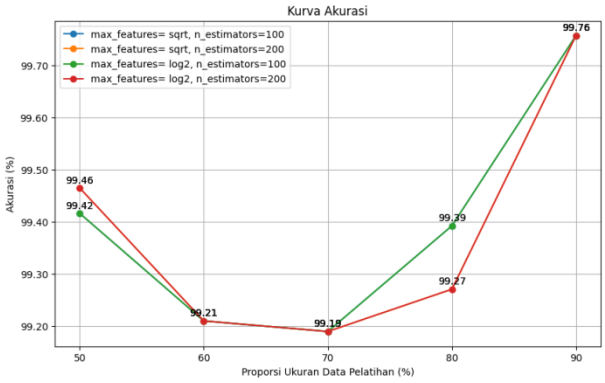
4.1.2 Uji Coba 2

Pada uji coba 2 menggunakan 4108 data dengan 3438 data kelas normal dan 670 data kelas MITM *attacking controller* (AC) dengan 5 pengujian seperti pada gambar 4.8 di bawah ini.

No.	Uji Coba 2	*Training*		*Testing*	
		Proporsi	Jumlah	Proporsi	Jumlah
1	Model 1	50%	2054	50%	2054
2	Model 2	60%	2464	40%	1644
3	Model 3	70%	2875	30%	1233
4	Model 4	80%	3286	20%	822
5	Model 5	90%	3697	10%	411

Gambar 4.8 Pembagian model pada uji coba 2

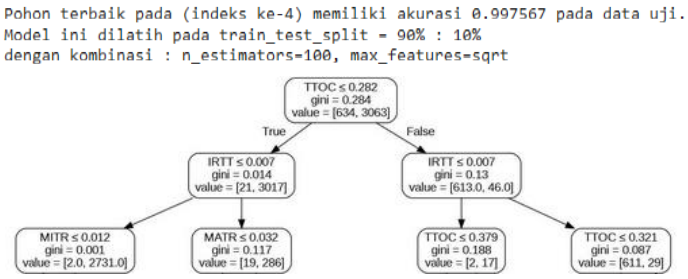
Terlihat pada gambar 4.8, uji coba ini memiliki 5 pengujian mulai dari model 1 menggunakan proporsi 50% data *training* dengan 2054 data dan 50% data *testing* dengan 2054 data hingga model 5 dengan proporsi 90% data *training* sebanyak 3697 data dan 10% data *testing* sebanyak 411 data. Di bawah ini ialah hasil pengujian dari 5 model.



Gambar 4.9 Kurva uji coba 2

Pada gambar 4.9 terlihat pada proporsi 50% : 50%, semua model menunjukkan akurasi yang cukup tinggi, dengan nilai maksimum 99.46% pada model dengan *max_features 'log2'* dan *n_estimators* 200. Akurasi sedikit menurun ketika proporsi data *training* meningkat ke 60% dan 70%. Menunjukkan bahwa model memiliki kesulitan mempertahankan performa saat data pelatihan bertambah pada proporsi ini. Akurasi mulai meningkat signifikan pada proporsi 80% dan mencapai puncaknya pada 90%, dengan *hyperparameter max_features 'sqrt'* dan *n_estimators* 100 mencapai akurasi tertinggi sebesar 99.76%.

Di bawah ini ialah gambar dari pohon terbaik pada uji coba 2.



Gambar 4.10 Pohon dengan akurasi terbaik pada uji coba 2

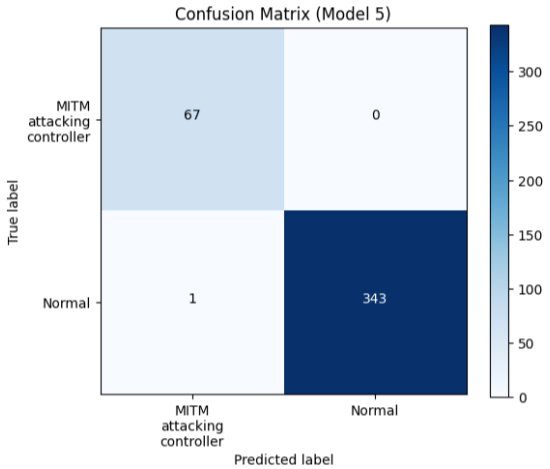
Gambar 4.10 merupakan informasi dan visualisasi pohon dengan akurasi terbaik pada uji coba 2. Pohon ini terletak pada indeks ke 4 dengan akurasi 99.75% pada data uji. Pohon ini berasal dari model 5 dengan proporsi 90% data latih dan 10% data uji. Pada *node root threshold* yang digunakan yaitu fitur TTOC dengan nilai 0.282, sehingga yang nilainya < *threshold* akan masuk leaf kiri selebihnya masuk leaf kanan. Dengan nilai *gini impurity* 0.284 menunjukan adanya campuran kelas. *Node* ini memiliki proporsi 634 data MITM AC dan 3063 data normal. Di bawah ini ialah hasil dari pengujian ini.

Train:Test = 90% : 10%

n_estimators	max_features	train_test_split	accuracy_test
100	sqrt	90% : 10%	0.997567
100	log2	90% : 10%	0.997567
200	sqrt	90% : 10%	0.997567
200	log2	90% : 10%	0.997567

Gambar 4.11 Hasil pengujian model

Hasil klasifikasi pada pengujian model ini dengan beberapa *hyperparameter* ditunjukkan pada gambar 4.11 dimana semua skenario memiliki nilai akurasi 99.76% sehingga model terbaiknya diambil pada model pertama dengan kombinasi *n_estimator* 100 dan *max_features* '*sqrt'*.




```
Model 5: 90% Train, 10% Test, n_estimators=100, max_features='sqrt'
Accuracy: 0.9975669099756691
Classification Report:
 Class  Precision    Recall     F1-Score    Support
   0   0.98529412  1.00000000  0.99259259     67.0
   1   1.00000000  0.99709302  0.99854440    344.0
 avg   0.99264706  0.99854651  0.99556849
True Positive: [67, 343]
True Negative: [343, 67]
False Positive: [1, 0]
False Negative: [0, 1]
Training Time: 0.5674111843109131 seconds
Testing Time: 0.0217592716217041 seconds
```

Gambar 4.12 *Confussion matrix* dan *classification report* 90% : 10%

Dari visualisasi dan hasil *classification report* pada gambar 4.12 diketahui nilai dari *precision*, *recall*, dan *f1-score* dari masing-masing kelas. Nilai TP dari kelas *MITM attacking router* dan kelas Normal ialah 67 dan 343. Dengan TN adalah 343 dan 67. FP ialah 1 dan 0 serta FN ialah 0 dan 1. Pengujian ini menggunakan waktu *training* sebesar 0.56 detik dan waktu *testing* sebesar 0.02 detik. Dengan menggunakan rumus pada sub bab 3.5 untuk mengukur nilai akurasi, presisi *recall*, dan *f1-score*, maka didapatkan tabel 4.3 dibawah ini.

Tabel 4.3 Hasil *confusion matrix* pengujian model uji coba 2

No.	**Kategori**	**TP**	**TN**	**FP**	**FN**	**Presisi**	***Recall***	***F1-Score***
1.	MITM *two-way attack*	67	343	1	0	98.52%	100%	99.26%
2.	Normal	343	67	0	1	100%	99.70%	99.85%
			Rata - rata			99.26%	99.85%	99.56%

$$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \; x \; 100\% = \frac{67 + 343}{411} \; x \; 100\% = 99.76\%$$

Pada tabel 4.3 didapatkan nilai rata-rata presisi, *recall*, *f1-score* dan akurasi pada setiap model pengujian. Di bawah ini merupakan nilai akurasi, presisi, *recall*, dan *f1-score* dari uji coba 2.

Tabel 4.4 Hasil performa pada uji coba 2

Model	***Training : Testing***	**Akurasi**	**Presisi**	***Recall***	***F1-Score***
1	50% : 50%	99.46%	98.63%	99.44%	99.03%
2	60% : 40%	99.21%	98.07%	99.08%	98.57%
3	70% : 30%	99.19%	98.15%	98.91%	98.53%
4	80% : 20%	99.39%	99.03%	98.74%	98.88%
5	90% : 10%	99.76%	99.26%	99.85%	99.56%

Terlihat pada tabel 4.4 bahwa akurasi tertinggi terdapat pada model 5 dengan nilai akurasi mencapai 99.76%, presisi 99.26%, *recall* 99.85%, dan *f1-score* 99.56%.

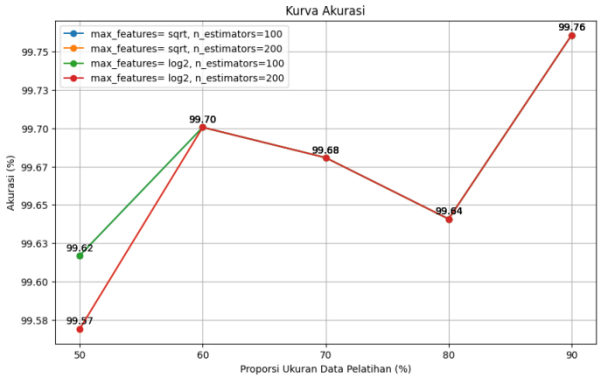
4.1.3 Uji Coba 3

Pada uji coba 3 menggunakan 4175 data dengan 3438 data kelas normal dan 737 data kelas MITM *attacking router* (AR) dengan 5 pengujian seperti pada gambar 4.13 di bawah ini.

No.	Uji Coba 3	*Training*		*Testing*	
		Proporsi	Jumlah	Proporsi	Jumlah
1	Model 1	50%	2087	50%	2088
2	Model 2	60%	2505	40%	1670
3	Model 3	70%	2922	30%	1253
4	Model 4	80%	3340	20%	835
5	Model 5	90%	3757	10%	418

Gambar 4.13 Pembagian model pada uji coba 3

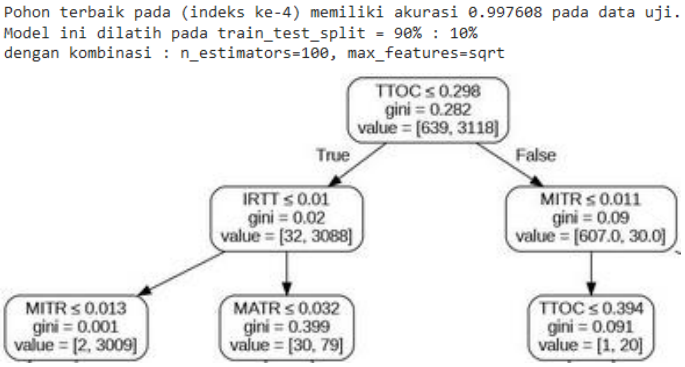
Terlihat pada gambar 4.13, uji coba ini memiliki 5 pengujian mulai dari model 1 menggunakan proporsi 50% data *training* sebanyak 2087 data dan 50% data *testing* sebanyak 2087 data hingga model 5 dengan proporsi 90% data *training* sebanyak 3757 data dan 10% data *testing* sebanyak 418 data. Di bawah ini ialah hasil pengujian dari 5 model pada uji coba ini.



Gambar 4.14 Kurva uji coba 3

Pada gambar 4.14 terlihat pada rasio 50%:50%, menghasilkan akurasi rata-rata di 99.59%. Akurasi meningkat secara signifikan pada rasio 60%:40%, terutama untuk kombinasi *hyperparameter max_features 'sqrt'* dan *n_estimator* 100, yang mencapai akurasi tertinggi pada proporsi ini sebesar 99.70%. Kemudian, akurasi model sedikit menurun relatif kecil (sekitar 0.02%-0.06%) pada rasio 70% : 30% dan 80% : 20%. Akurasi kembali meningkat secara signifikan pada rasio 90% : 10%, dengan kombinasi *max_features 'sqrt'* dan *n_estimators* 100 menghasilkan akurasi tertinggi sebesar 99.76%.

Di bawah ini ialah gambar dari pohon terbaik pada uji coba 3.



Gambar 4.15 Pohon dengan akurasi terbaik pada uji coba 3

Gambar 4.15 merupakan informasi dan visualisasi pohon dengan akurasi terbaik pada uji coba 3. Pohon ini terletak pada indeks ke 4 dengan akurasi 99.76% pada data uji. Pohon ini berasal dari model 5 dengan proporsi 90% data *training* dan 10% data *testing* dengan *n_estimators* 100 dan *max_features 'sqrt'*. Pada *node root*, *threshold* yang digunakan yaitu fitur TTOC dengan nilai 0.298, sehingga yang nilainya < *threshold* akan masuk leaf kiri selebihnya masuk leaf kanan. Dengan nilai *gini impurity* 0.282 menunjukan adanya campuran kelas. *Node* ini memiliki proporsi 639 data MITM AC dan 3118 data normal. Di bawah ini ialah hasil dari pengujian ini.

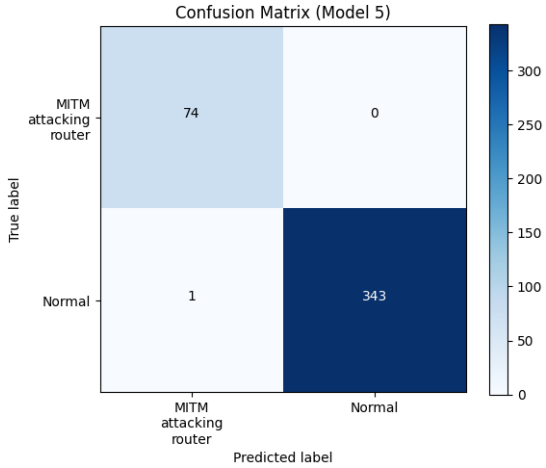
Train:Test = 90% : 10%

n_estimators	max_features	train_test_split	accuracy_test
100	sqrt	90% : 10%	0.997608
100	log2	90% : 10%	0.997608
200	sqrt	90% : 10%	0.997608
200	log2	90% : 10%	0.997608

Gambar 4.16 Hasil pengujian model

Hasil klasifikasi pada pengujian model ini dengan beberapa *hyperparameter* ditunjukkan pada gambar 4.16 dimana semua skenario memiliki

nilai akurasi 99.76% sehingga model terbaiknya diambil pada model pertama dengan kombinasi *n_estimator* 100, *max_features* '*sqrt'*.




```
Model 5: 90% Train, 10% Test, n_estimators=100, max_features='sqrt'
Accuracy: 0.9976076555023924
Classification Report:
 Class  Precision    Recall     F1-Score    Support
   0   0.98666667  1.00000000  0.99328859     74.0
   1   1.00000000  0.99709302  0.99854440    344.0
 avg   0.99333333  0.99854651  0.99591649
True Positive: [74, 343]
True Negative: [343, 74]
False Positive: [1, 0]
False Negative: [0, 1]
Training Time: 0.6314489841461182 seconds
Testing Time: 0.03041815757751465 seconds
```

Gambar 4.17 *Confussion matrix* dan *classification report* 90% : 10%

Dari visualisasi dan hasil *classification report* pada gambar 4.17 diketahui nilai dari *precision*, *recall*, dan *f1-score* dari masing-masing kelas. Nilai TP dari dari kelas *MITM attacking router*, dan kelas Normal ialah 74 dan 343. Dengan TN adalah 343 dan 74. FP ialah 1 dan 0 serta FN ialah 0 dan 1. Pengujian ini menggunakan waktu *training* sebesar 0.63 detik dan waktu *testing* sebesar 0.03 detik. Dengan menggunakan rumus pada sub bab 3.5 untuk mengukur nilai akurasi, presisi *recall*, dan *f1-score*, maka didapatkan tabel 4.5 dibawah ini.

Tabel 4.5 Hasil *confusion matrix* pengujian model uji coba 3

No.	Kategori	TP	TN	FP	FN	Presisi	*Recall*	*F1-Score*
1.	MITM *two-way attack*	73	343	1	0	98.66%	100%	99.33%
2.	Normal	343	73	0	1	100%	99.70%	99.85%
			Rata - rata			99.33%	99.85%	99.59%

$$\text{Akurasi} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \; x \; 100\% = \frac{74 + 343}{418} \; x \; 100\% = 99.76\%$$

Pada tabel 4.5 didapatkan nilai rata-rata presisi, *recall, f1-score* dan akurasi pada setiap model pengujian. Di bawah ini merupakan nilai akurasi, presisi, *recall,* dan *f1-score* dari uji coba 3.

Tabel 4.6 Hasil performa pada uji coba 3

Model	*Training : Testing*	Akurasi	Presisi	*Recall*	*F1-Score*
1	50% : 50%	99.62%	99.04%	99.66%	99.35%
2	60% : 40%	99.70%	99.29%	99.69%	99.49%
3	70% : 30%	99.68%	99.11%	99.81%	99.45%
4	80% : 20%	99.64%	99.00%	99.78%	99.39%
5	90% : 10%	99.76%	99.33%	99.85%	99.59%

Terlihat pada tabel 4.6 bahwa akurasi tertinggi terdapat pada model 5 dengan nilai akurasi mencapai 99.76%, presisi 99.33%, *recall* 99.85%, dan *f1-score* mencapai 99.59%.

4.3 Pembahasan

Berdasarkan hasil uji coba yang sudah dilakukan pada sub-bab 4.2 dengan 3 uji coba dengan masing-masing 5 skenario berbeda, uji coba pertama dengan kelas Normal dan MITM *two-way attack* menghasilkan akurasi pada proporsi

train : test 50% : 50% sebesar 99.95% dan meningkat seiring pertambahan proporsi data *training*. Uji coba 1 mencapai akurasi presisi, *recall*, dan *f1-score* tertinggi sebesar 100% pada proporsi data 60% : 40% dan stabil hingga 90% : 10%. Hal ini menunjukkan bahwa uji coba 1 mampu memprediksi semua data uji dengan benar.

Uji coba kedua dengan kelas Normal dan MITM *attacking router* menghasilkan akurasi pada proporsi *train* 50% sebesar 99.46% dengan kombinasi *n_estimators* 200 dan *max_features* '*sqrt'*. Akurasi sedikit menurun ketika proporsi data *training* meningkat ke 60% dan 70%. Menunjukkan bahwa model mengalami fluktuasi dalam mempertahankan performa saat data pelatihan bertambah pada proporsi ini. Akurasi mulai meningkat signifikan pada proporsi 80% dan mencapai puncaknya pada 90%, dengan kombinasi *max_features 'sqrt'* dan *n_estimator* 100 mencapai akurasi tertinggi sebesar 99.76% dengan nilai presisi 99.26%, *recall* 99.85% dan *f1-score* 99.56%.

Uji coba ketiga dengan kelas Normal dan MITM *attacking controller* menghasilkan akurasi pada proporsi *train* 50% sebesar 99.62% dengan kombinasi *n_estimators* 100 dan *max_features* '*sqrt'*. Akurasi meningkat secara signifikan pada rasio 60%:40%, terutama untuk kombinasi hyperparameter *max_features 'sqrt'* dan n_estimators 100, yang mencapai akurasi tertinggi pada proporsi ini sebesar 99.70%. Kemudian, akurasi model sedikit menurun relatif kecil (sekitar 0.02%-0.06%) pada rasio 70%:30% dan 80%:20%. Akurasi kembali meningkat secara signifikan pada rasio 90%:10%, dengan kombinasi *max_features 'sqrt'* dan

n_estimator 100 menghasilkan akurasi tertinggi pada uji coba 3 sebesar 99.76% dengan nilai presisi 99.33%, *recall* 99.85% dan *f1-score* 99.59%.

Dari hasil evaluasi ketiga uji coba, menunjukkan bahwa algoritma *Random Forest* berhasil melakukan analisa dan mampu membedakan antara data serangan dan data normal. Uji coba 1, dengan rasio data *training* 60% hingga 90% menghasilkan nilai akurasi, presisi, *recall*, dan *f1-score* sebesar 100% untuk semua kombinasi. Sehingga pada uji coba 1 dengan data Normal dan MITM *two way attack*, algoritma mampu memprediksi dengan sangat baik. Pada uji coba 2 mencapai performa tertinggi pada rasio data *training* 90% dengan *n_estimator* 100 dan *max_featrues 'sqrt*' menghasilkan akurasi sebesar 99.76%, presisi 99.26%, *recall* 99.85%, dan *f1-score* 99.86%. Sehingga pada uji coba 2 dengan data Normal dan MITM *attacking controller* algoritma mampu memprediksi dengan sangat baik. Kemudian uji coba 3 mencapai performa tertinggi pada rasio data *training* 90% dengan *n_estimator* 100 dan *max_featrues 'sqrt*' menghasilkan akurasi sebesar 99.76%, presisi 99.33%, *recall* 99.85%, dan *f1-score* 99.59%. Sehingga pada uji coba 3 dengan data Normal dan MITM *attacking router* algoritma mampu memprediksi dengan sangat baik. Hal ini menunjukkan bahwa model deteksi serangan *Man In The Middle* (MITM) pada protokol Modbus TCP/IP di jaringan *Smart Grid* menghasilkan nilai akurasi yang sangat baik.

Penelitian ini sesuai dengan konsep muamalah, yaitu muamalah mu'Allah yang merupakan segala sesuatu yang berhubungan dengan Allah. Muamalah mu'Annas yang merupakan hal-hal yang berhubungan dengan sesama manusia. Dan muamalah mu'Allam yang merupakan hal-hal yang berhubungan dengan

alam. Perspektif islam mengenai klasifikasi tidak dibahas secara langsung, pada muamalah mu'Allah penelitian ini sesuai dengan firman Allah SWT mengenai pemisahan antara kebenaran dan kebatilan dengan konsep klasifikasi pada penelitian ini. Allah SWT berfirman dalam QS *Ali 'Imran* ayat 179:

"مَا كَانَ ٱللَّهُ لِيَذَرَ ٱلْمُؤْمِنِينَ عَلَىٰ مَآ أَنتُمْ عَلَيْهِ حَتَّىٰ يَمِيزَ ٱلْخَبِيثَ مِنَ ٱلطَّيِّبِ ۗ وَمَا كَانَ ٱللَّهُ لِيُطْلِعَكُمْ عَلَى ٱلْغَيْبِ وَلَٰكِنَّ ٱللَّهَ يَجْتَبِى مِن رُّسُلِهِۦ مَن يَشَآءُ ۖ فَـَٔامِنُوا۟ بِٱللَّهِ وَرُسُلِهِۦ ۚ وَإِن تُؤْمِنُوا۟ وَتَتَّقُوا۟ فَلَكُمْ أَجْرٌ عَظِيمٌ"

"Allah tidak akan membiarkan orang-orang yang beriman dalam keadaan kamu sekarang ini, sehingga Dia memisahkan yang buruk dari yang baik. Dan Allah tidak akan memperlihatkan kepadamu hal-hal yang gaib, tetapi Allah memilih siapa yang Dia kehendaki di antara rasul-rasul-Nya. Karena itu berimanlah kepada Allah dan rasul-rasul-Nya; dan jika kamu beriman dan bertakwa, maka bagimu pahala yang besar." (QS Ali 'Imran:179)

Menurut tafsir dari Kemenag (Kementerian Agama RI), Sebagai bagian dari ketentuan Allah bagi hamba-Nya, Dia tidak akan membiarkan orang-orang beriman terus-menerus berada dalam kesulitan, seperti yang terjadi pada Perang Uhud. Allah tidak membiarkan campur aduk antara orang mukmin dan orang munafik, tetapi membedakan keduanya melalui wahyu yang diturunkan kepada Nabi Muhammad. Allah tidak memperlihatkan hal-hal yang ghaib kepada umat manusia, namun Dia memilih siapa yang Dia kehendaki di antara rasul-rasul-Nya dengan pengetahuan yang mendalam tentang isi hati manusia, sehingga Dia mengetahui siapa yang beriman dan siapa yang munafik. Oleh karena itu, berimanlah kepada Allah dan rasul-rasul-Nya. Jika kamu beriman dan bertakwa dengan melaksanakan perintah-Nya dan menjauhi larangan-Nya, maka kamu akan mendapatkan pahala yang besar di sisi-Nya bersama para kekasih-Nya di surga yang penuh kenikmatan (Kemenag, 2022).

Dalam hadits shahih riwayat Muslim no 2699 Rasulullah SAW bersabda:

وَمَنْ سَلَكَ طَرِيقًا يَلْتَمِسُ فِيهِ عِلْمًا سَهَّلَ اللَّهُ لَهُ بِهِ طَرِيقًا إِلَى الْجَنَّةِ

"Barang siapa yang menempuh jalan untuk mencari ilmu, maka Allah akan mudahkan baginya jalan menuju surga." (HR. Muslim, no. 2699).

Berdasarkan tafsir ayat diatas Allah SWT tidak akan membiarkan orang-orang beriman dalam kebingungan hingga Allah memisahkan orang yang buruk dari yang baik. Dipertegas dengan hadits diatas bahwa usaha dalam mencari ilmu adalah jalan menuju ridha Allah. Dalam penelitian ini, klasifikasi merupakan pemisahan yang baik dari yang buruk yang dapat diartikan sebagai upaya manusia dalam menggunakan ilmu pengetahuan untuk memisahkan hal baik (data normal) dari yang buruk (data serangan) dalam mendeteksi serangan pada protokol *Modbus* di jaringan *Smart Grid*. Hal ini mencerminkan ketaatan dan pengabdian kepada Allah dengan memanfaatkan ilmu yang bermanfaat bagi banyak orang dan menjaga keamanan.

Pengelolaan informasi yang aman serta pencegahan akses tanpa izin merupakan bagian dari menjaga hak-hak sesama manusia. Penelitian ini diharapkan dapat melindungi hak pengguna dari serangan yang bisa merusak atau mencuri data penting. Hal ini sesuai dengan Muamalah mu'Annas. Allah SWT berfirman dalam QS *An-Nur* ayat 27:

"يَٰٓأَيُّهَا ٱلَّذِينَ ءَامَنُواْ لَا تَدۡخُلُواْ بُيُوتًا غَيۡرَ بُيُوتِكُمۡ حَتَّىٰ تَسۡتَأۡنِسُواْ وَتُسَلِّمُواْ عَلَىٰٓ أَهۡلِهَاۖ ذَٰلِكُمۡ خَيۡرٞ لَّكُمۡ لَعَلَّكُمۡ تَذَكَّرُونَ"

"Wahai orang-orang yang beriman! Janganlah kamu memasuki rumah yang bukan rumahmu sebelum meminta izin dan memberi salam kepada penghuninya. Yang demikian itu lebih baik bagimu, agar kamu selalu ingat." (QS An-Nur: 27)

Berdasarkan tafsir Kemenag, ayat ini mengungkapkan etika kunjungan untuk menjaga cinta, kasih sayang, serta hubungan yang harmonis di antara mereka, dengan jangan memasuki rumah orang lain kecuali setelah mendapat izin dan mengucapkan salam terlebih dahulu, agar tidak melihat hal-hal yang seharusnya tidak diketahui orang lain. Jika seseorang meminta izin untuk masuk, yang ditandai dengan memberikan salam, namun tidak ada jawaban, maka sebaiknya permintaan itu diulang hingga tiga kali. Setelah mendapatkan izin, barulah diperbolehkan masuk, jika tidak, disarankan untuk pergi (Kemenag, 2022).

Dalam hadits shahih Rasulullah SAW bersabda:

خَيْرُ الناسِ أَنفَعُهُم لِلنَّاسِ

"Sebaik-baik manusia adalah yang paling bermanfaat bagi orang lain." (HR ath-Thabrani, Al-Mu'jam al-Ausath, juz VII, hal. 58).

Ayat diatas mengajarkan pentingnya menghormati batasan privasi. Seseorang harus meminta izin sebelum memasuki area milik orang lain. Dengan kata lain, Islam menekankan pentingnya menghormati hak dan privasi sesama. Ayat ini relevan dengan penelitian ini karena deteksi serangan dalam jaringan berfungsi untuk melindungi data dari akses tidak sah, yang dapat mengganggu hak-hak pengguna. Hal ini ditegaskan dengan hadis diatas yang menerangkan bahwa manusia yang terbaik adalah mereka yang memberikan manfaat paling besar bagi orang lain. Sehingga dengan mendeteksi dan mencegah serangan pada jaringan, penelitian ini memberikan manfaat besar bagi masyarakat dalam melindungi data dan privasi. Serta menjaga hak-hak privasi orang lain dan

mencegah kerugian yang mungkin terjadi akibat akses ilegal maupun pencurian data.

Penelitian ini juga berkaitan dengan pengelolaan sumber daya energi listrik dalam jaringan *Smart Grid* yang sesuai dengan konsep muamalah mua'allam. Sebagai bagian dari alam, energi adalah sumber daya yang harus dikelola dengan baik agar tidak rusak dan disalahgunakan. MITM adalah jenis serangan yang sangat merugikan. Serangan MITM terhadap protokol Modbus TCP/IP pada *Smart Grid* dapat menyebabkan pencurian data, pemalsuan data, bahkan gangguan terhadap operasi jaringan secara keseluruhan. Jika hal ini terjadi maka akan mengakibatkan kerugian & kerusakan yang besar.

Sebagaimana dalam QS *Al A'raf* ayat 56:

وَلَا تُفْسِدُوْا فِى الْاَرْضِ بَعْدَ اِصْلَاحِهَا وَادْعُوْهُ خَوْفًا وَّطَمَعًاۗ اِنَّ رَحْمَتَ اللّٰهِ قَرِيْبٌ مِّنَ الْمُحْسِنِيْنَ

"Janganlah kamu berbuat kerusakan di bumi setelah diatur dengan baik. Berdoalah kepada-Nya dengan rasa takut dan penuh harap. Sesungguhnya rahmat Allah sangat dekat dengan orang-orang yang berbuat baik." (QS Al-A'raf: 56).

Dijelaskan dalam Tafsir Kemenag, Allah melarang manusia pada ayat ini, agar tidak membuat kerusakan di bumi. Larangan berbuat kerusakan ini berlaku dalam berbagai aspek, seperti merusak hubungan sosial, kesehatan fisik dan mental orang lain, kehidupan serta sumber daya yang mendukung kehidupan. Bumi ini diciptakan oleh Allah dengan segala kelengkapannya, seperti lembah, gunung, daratan, lautan, hutan, sungai dan lain-lain, yang semuanya dimanfaatkan dengan sebaik-baiknya untuk kesejahteraannya manusia. Oleh karena itu, manusia dilarang membuat kerusakan di bumi (Kemenag, 2022).

Hal ini dipertegas dengan hadits di bawah ini:

عَنْ أَبِي سَعِيْدٍ سَعْدِ بْنِ مَالِكِ بْنِ سِنَانٍ الخُدْرِيِّ رَضِيَ اللهُ عَنْهُ أَنَّ رَسُوْلَ اللهِ ﷺقَالَ: «لاَ ضَرَرَ وَلاَ ضِرَارَ»حَدِيْثٌ حَسَنٌ. رَوَاهُ ابْنُ مَاجَهْ وَالدَّارَقُطْنِيُّ وَغَيْرُهُمَا مُسْنَدًا، وَرَوَاهُ مَالِكٌ فِي المؤَطَّأِ مُرْسَلاً عَنْ عَمْرِو بْنِ يَحْيَى عَنْ أَبِيْهِ عَنِ النَّبِيِّ ﷺفَأَسْقَطَ أَبَا سَعِيْدٍ، وَلَهُ طُرُقٌ يُقَوِّي بَعْضُهَا بَعْضًا

Dari Abu Sa'id Al-Khudri RA bahwa Rasulullah SAW bersabda, "Tidak boleh memberikan mudarat tanpa disengaja atau pun disengaja." (HR. Ibnu Majah, no. 2340 dan 2341)

Serangan *Man in The Middle* merupakan tindakan yang merusak dan dilarang dalam Al-Qur'an. Surat *Al-A'raf* ayat 56 menegaskan larangan berbuat segala bentuk kerusakan yang dapat menimbulkan kerugian. Hadis diatas juga menegaskan bahwa segala bentuk kerusakan, baik yang membahayakan diri sendiri maupun orang lain, harus dihindari. Serangan *Man in The Middle* pada *Smart Grid*, dengan dampaknya yang dapat menyebabkan pencurian data, pemalsuan data, bahkan gangguan terhadap operasi jaringan secara keseluruhan, dapat dilihat sebagai salah satu bentuk kerusakan di muka bumi. Oleh karena itu, hal ini mengingatkan kita untuk tidak menyebabkan kerusakan, namun sebaliknya, untuk berusaha melakukan kebaikan dan menjaga stabilitas dalam segala hal, dengan melalui deteksi serangan *Man in The Middle* pada jaringan *Smart Grid* termasuk dalam menjaga kebaikan dalam penggunaan layanan dan teknologi.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa algoritma *Random Forest* terbukti efektif dalam mengklasifikasikan data normal dan serangan pada protokol Modbus TCP/IP di jaringan *Smart Grid*. Hal ini dibuktikan melalui tiga kali uji coba dengan uji coba 1 menggunakan data Normal dan MITM *two way attack*, uji coba 2 menggunakan data Normal dan MITM *attacking controller* dan uji coba 3 menggunakan data Normal dan MITM *attacking router*. Setiap uji coba dilakukan lima kali pengujian dengan komposisi data latih dan uji yang berbeda, yaitu 50%:50%, 60%:40%, 70%:30%, 80%:20% dan 90%:10%. Hasil dari ketiga uji coba menunjukkan *Random Forest* mampu mencapai rata-rata akurasi di atas 99%. Dengan hasil terbaik pada uji coba 1 pada proporsi data latih:data uji 60%:40% hingga 90%:10% yang menghasilkan akurasi 100%, presisi 100%, *recall* 100% dan *f1-score* sebesar 100%.

5.2 Saran

Untuk pengembangan pada penelitian selanjutnya disarankan untuk mempertimbangkan *tuning hyperparameter*. Metode *Grid Search* atau *Random Search* dapat digunakan untuk melakukan *tuning hyperparameter*. Metode ini dapat digunakan untuk membantu menemukan kombinasi *hyperparameter* terbaik yang sesuai dengan karakteristik data yang digunakan. Sehingga diharapkan dapat meningkatkan performa pada algoritma dan hasil penelitian.

DAFTAR PUSTAKA

Durairaj, D., Wróblewski, Ł., Sheela, A., Hariharasudan, A., & Urbański, M. (2022). Random forest based power sustainability and cost optimization in smart grid. *Production Engineering Archives*, *28*(1), 82–92. https://doi.org/10.30657/pea.2022.28.10

Ekawijana, A., Bakhrun, A., & Kurniawan, M. T. (2024). *Deteksi Serangan DDOS Pada Jaringan SDN dengan Metode Random Forest*. *8*.

Elbez, G., Keller, H. B., & Hagenmeyer, V. (2018). A New Classification of Attacks against the Cyber-Physical Security of Smart Grids. *Proceedings of the 13th International Conference on Availability, Reliability and Security*, 1–6. Hamburg Germany: ACM. https://doi.org/10.1145/3230833.3234689

Elrawy, M. F., Hadjidemetriou, L., Laoudias, C., & Michael, M. K. (2023). Detecting and classifying man-in-the-middle attacks in the private area network of smart grids. *Sustainable Energy, Grids and Networks*, *36*, 101167. https://doi.org/10.1016/j.segan.2023.101167

Farhan, M. R. (2021). *Sistem Deteksi Man In The Middle (MITM) Attack Pada Jaringan Upervisory Control And Data Acquisition (SCADA) Dengan Menggunakan Decision Tree* (Universitas Sriwijaya). Universitas Sriwijaya, Palembang. Retrieved from https://repository.unsri.ac.id/62953/

Ghosh, A., & Kole, A. (2021, October 26). *A Comparative Analysis of Enhanced Machine Learning Algorithms for Smart Grid Stability Prediction*. https://doi.org/10.36227/techrxiv.16863145.v1

Ghosh, T., Bagui, S., Bagui, S., Kadzis, M., & Bare, J. (2023). Anomaly Detection for Modbus over TCP in Control Systems Using Entropy and Classification-Based Analysis. *Journal of Cybersecurity and Privacy*, *3*(4), 895–913. https://doi.org/10.3390/jcp3040041

Kemenag. (2022). *Tafsir Ringkas Kemenag & Tafsir Tahlili Al Qur'an*. Tafsir. Retrieved from https://quran.kemenag.go.id/quran/per-ayat/surah

Khan, S., Kifayat, K., Kashif Bashir, A., Gurtov, A., & Hassan, M. (2021). Intelligent intrusion detection system in smart grid using computational intelligence and machine learning. *Transactions on Emerging Telecommunications Technologies*, *32*(6), e4062. https://doi.org/10.1002/ett.4062

Kurniabudi, K., Harris, A., & Rahim, A. (2020). Seleksi Fitur Dengan Information Gain Untuk Meningkatkan Deteksi Serangan DDoS menggunakan Random Forest. *Techno.Com*, *19*(1), 56–66. https://doi.org/10.33633/tc.v19i1.2860

Prakasa, J. E. W. (2020). Peningkatan Keamanan Sistem Informasi Melalui Klasifikasi Serangan Terhadap Sistem Informasi. *Jurnal Ilmiah Teknologi Informasi Asia*, *14*(2), 75. https://doi.org/10.32815/jitika.v14i2.452

Prima, A. S. (2022). *Klasifikasi Serangan Man In The Middle (MITM) Pada Protokol Jaringan SCADA (Iec 60870-5-104) Menggunakan Metode Logistic Regression* (Universitas Sriwijaya). Universitas Sriwijaya, Palembang. Retrieved from https://repository.unsri.ac.id/75393/

Qu, H., Qin, J., Liu, W., & Chen, H. (2018). Instruction Detection in SCADA/Modbus Network Based on Machine Learning. In X. Gu, G. Liu, & B. Li (Eds.), *Machine Learning and Intelligent Communications* (pp. 437–454). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-73447-7_48

Qu, Z., Zhang, X., Gao, Y., Peng, C., Wang, Y., & Georgievitch, P. M. (2023). Detection of False Data Injection Attack in AGC System Based on Random Forest. *Machines*, *11*(1), 83. https://doi.org/10.3390/machines11010083

Rafsanjani, M. S., Suryani, V., & Pahlevi, R. R. (2022). *Deteksi Serangan Botnet Pada Jaringan Internet of Things Menggunakan Algoritma Random Forest (RF)*.

Septiano, S. (2021). *Deteksi Serangan Man In The Middle (MITM) Attack Pada Jaringan Supervisory Control And Data Acquisition (SCADA) Menggunakan Random Forest* (Universitas Sriwijaya). Universitas Sriwijaya, Palembang. Retrieved from https://repository.unsri.ac.id/48283/

Setiadi, R. R. (2021). *Implementasi dan Deteksi Serangan Man-In-The-Middle Berbasis MITM Proxy Terhadap Protokol HTTPS Menggunakan Metode K-NN*. Retrieved from https://openlibrarypublications.telkomuniversity.ac.id/index.php/engineering/article/view/15681

Sholihah, N. N., & Hermawan, A. (2023). Implementation of Random Forest and Smote Methods for Economic Status Classification in Cirebon City. *Jurnal Teknik Informatika (Jutif)*, *4*(6), 1387–1397. https://doi.org/10.52436/1.jutif.2023.4.6.1135

Sinaga, D. H., Sasue, R. R. O., & Hutahaean, H. D. (2021). Pemanfaatan Energi Terbarukan Dengan Menerapkan Smart Grid Sebagai Jaringan Listrik Masa Depan. *Journal Zetroem*, *3*(1), 11–17. https://doi.org/10.36526/ztr.v3i1.1251

Uzlah, L. I., & Saputra, R. A. (2024). *Deteksi Serangan Siber Pada Jaringan Komputer Menggunakan Metode Random Forest*. *8*(3).

Ye, Y., Wu, Q., Zhexue Huang, J., Ng, M. K., & Li, X. (2013). Stratified sampling for feature subspace selection in random forests for high dimensional data. *Pattern Recognition*, *46*(3), 769–787. https://doi.org/10.1016/j.patcog.2012.09.005