Aplikasi *Enterprise Document Digital Signature* menggunakan RSA dan SHA256 untuk WFH di Era Pandemi COVID-19

Rafie Afif Andika[1], Aji Gautama Putradana[2], Rizka Reza Pahlevi[3]

[1,2,3] Universitas Telkom, Bandung
rafieafifandika@students.telkomuniversity.ac.id[1], aji@telkomuniversity.ac.id[2], reza@telkomuniversity.ac.id[3]

Abstrak

Dalam situasi WFH ditengah Pandemi COVID-19 dimana beberapa hal dikerjakan dengan pemanfaatan internet, aktivitas seperti mengirim dokumen secara online dengan scan tanda tangan sudah menjadi hal yang standar. Namun, tanda tangan yang dipindai dapat dengan mudah dipalsukan, dicuri, dan disalahgunakan. Penelitian ini bertujuan untuk membuat dan mengimplementasikan aplikasi tanda tangan digital dokumen perusahaan menggunakan RSA dan SHA256 sehingga sistem WFH di era pandemi COVID-19 dapat terselenggara dengan efektif dan aman. Sebagai bukti konsep, aplikasi dummy untuk perusahaan dibuat. Aplikasi ini merupakan sebuah sistem terdistribusi yang dapat berbagi dokumen untuk pemegang dokumen (pemohon), pemegang tanda tangan (signer), dan pemverifikasi tanda tangan dan dokumen (verifier). Dua skenario tanda tangan digital dibuat untuk perbandingan, satu menggunakan enkripsi RSA 2048 bit dan yang lainnya menggunakan enkripsi RSA 4098 bit. Dari hasil pengujian overhead RSA 4096 bit membutuhkan waktu kurang lebih 4 kali dari waktu overhead RSA 2048 bit untuk proses signature dan proses verifikasi. Namun, melalui perhitungan simulasi brute force, RSA 4096 bit membutuhkan sekitar 10^{616} kali lebih lama untuk diretas dibandingkan dengan RSA 2048 bit. Selain itu, melalui uji integritas, verifier dapat mendeteksi jika dokumen atau kunci tanda tangan apabila dipalsukan.

Kata kunci : ***digital signature, SHA256, RSA, enterprise, COVID-19, brute force attack***

Abstract

In a COVID-19 Pandemic situation of WFH with the use of the internet, activities such as sending documents online with scanned signature have become standard. However, scanned signature can be easily forged, stolen, and misused. The aim of this research is to create and implement an application for enterprise document digital signatures using RSA and SHA256 so that WFH in COVID-19 pandemic era can be held effective and securely. For proof of concept, an enterprise-like dummy application is created. The application is a distributed system that can share documents for the document holder (applicant), the signature holder (signer), and the signature and document verifier (verifier). Two digital signature scenarios are created for comparison, one uses RSA 2048 bit encryption and the other uses RSA 4098 bit encryption. From overhead testing results, RSA 4096 bit requires approximately 4 times the overhead time of RSA 2048 bit for the signature process and the verification process. However, through calculation brute force simulations, RSA 4096 bit requires approximately 10^{616} times longer to crack compared to RSA 2048 bit. Also, through integrity test, the verifier can detect if the document or the signature key is falsified.

Keywords: ***digital signature, SHA256, RSA, enterprise, COVID-19, brute force attack***

1. Pendahuluan

1.1 Latar Belakang

Corona Virus Disease 2019 (COVID-19) merupakan sebuah pandemi yang menyebar dengan cepat dan global, serta terjadi dalam jangka waktu yang lama [1] [2]. Salah satu konsekuensinya adalah penerapan sistem yang disebut *work from home* (WFH) yang diberlakukan oleh pemerintah [3], di mana penerapannya adalah dengan melakukan pekerjaan dari rumah oleh setiap perusahaan. Sebagai salah satu hasil WFH, menurut Kementerian Komunikasi dan Informatika Republik Indonesia (KOMINFO), penggunaan internet meningkat 40% [4]. Dalam situasi tersebut segala pekerjaan dikerjakan dengan menggunakan internet, aktivitas seperti mengirim dokumen secara online sudah menjadi standar [5]. Salah satu risiko yang ditimbulkan dari pengiriman dokumen saat WFH adalah pemalsuan dokumen[6]. Salah satu cara pemalsuan dokumen adalah dengan pemalsuan tanda tangan [7]. Oleh karena itu, diperlukan suatu sistem keamanan yang dapat menjamin keutuhan suatu dokumen digital perusahaan [8].

Dalam praktik WFH, memberikan tanda tangan dalam bentuk fisik merupakan sesuatu yang tidak memungkinkan, sehingga tanda tangan yang dipindai telah banyak digunakan. Namun tanda tangan yang dipindai dapat dengan mudah dipalsukan, dicuri, dan disalahgunakan [9]. Tanda tangan digital merupakan salah satu

metode kriptografi yang dapat menjamin keutuhan suatu pesan atau dokumen [10] [11]. Metode ini bekerja dengan memberikan kode yang mewakili integritas suatu data [12]. Kode diberikan dalam dua tahap, tahap pertama membuat hash data, tahap kedua menjalankan enkripsi pada hash [13]. Setelah proses dilalui, perubahan data dapat dideteksi ketika penerima pesan memverifikasi tanda tangan digital [14].

Implementasi tanda tangan digital biasanya menggunakan kriptografi asimetris, di mana kunci privat berbeda dengan kunci public [15]. Metode RSA merupakan salah satu algoritma kriptografi asimetris yang dapat digunakan dalam aplikasi tanda tangan digital [16]. Kelebihan RSA adalah seperti yang telah disebutkan, dengan penggunaan kunci privat, RSA memecahkan masalah kerahasiaan dan keaslian kriptografi simetris [17].

Dalam proses tanda tangan digital, selain enkripsi, juga diperlukan proses *hashing* pada dokumen [18]. Salah satu metode yang dapat diusulkan untuk *hashing* dokumen yaitu dengan pemanfaatan variasi dari Secure Hash Algorithm (SHA) yaitu SHA256 [19]. Keuntungan dari SHA256 adalah metode *hashing* ini aman dan efisien [20]. Aplikasi tanda tangan digital dokumen perusahaan menggunakan RSA dan SHA256 sehingga WFH di era pandemi COVID-19 dapat terselenggara dengan efektif dan aman. Sebagai bukti konsep, aplikasi *dummy* seperti perusahaan dibuat. Aplikasi ini merupakan sistem terdistribusi yang dapat berbagi dokumen untuk pemegang dokumen (pemohon), pemegang tanda tangan (*signer*), dan pemverifikasi tanda tangan dan dokumen (*verifier*).

Makalah ini membahas tentang analisis keamanan *integrity* and *confidentiality* dengan membahas ketahanan kunci RSA 2048 bit dan 4096 bit terhadap serangan *brute force*. Analisis *overhead* diambil dari lamanya waktu untuk proses tanda tangan dan verifikasi dokumen.

1.2 Topik dan Batasannya

Berdasarkan latar belakang masalah yang telah dipaparkan, maka rumusan masalah yang diangkat oleh penulis pada penelitian ini yaitu bagaimana cara memberikan keamanan dokumen menggunakan algoritma RSA dan SHA256. Serta bagaimana menganalisa aspek keamanan dan *overhead* dari penerapan algoritma RSA dan SHA256 pada penelitian ini. Pada proses enkripsi dan dekripsi, jenis algoritma RSA yang digunakan yaitu RSA 2048 bit atau 4096 bit. Untuk proses *hashing*, algoritma yang digunakan yaitu SHA256. Metode yang digunakan pada penelitian ini diterapkan berbasiskan Web. Adapun parameter keamanan yang diukur yaitu *confidentiality* dan *integrity*. *Overhead analysis* pada penelitian ini diukur dari segi, lama waktu *signaturing* dan *verification* dokumen.

1.3 Tujuan

Tugas akhir ini bertujuan untuk memberikan keamanan dokumen dengan mempertimbangkan parameter keamanan yaitu *confidentiality* dan *integrity*. Adapun keberhasilan parameter *integrity* dilihat dari apakah terdeteksinya perubahan yang ada pada sebuah dokumen. Sedangkan keberhasilan parameter *confidentiality* diukur dengan melakukan simulasi perhitungan *brute force attack* terhadap ketahanan RSA 2048 bit dan 4096 bit.

1.4 Organisasi Tulisan

Buku tugas akhir ini disusun menjadi lima bagian yaitu Bab 1 – Pendahuluan yang berisikan latar belakang, topik dan batasannya, serta tujuan penelitian. Bab 2 – Studi terkait berisi studi literatur atau penelitian terdahulu yang dijadikan sebagai acuan dalam penelitian ini. Bab 3 – Perancangan Sistem, bab ini berisi alur sistem yang dikerjakan pada penelitian ini. Bab 4 – Implementasi dan Evaluasi menjelaskan tentang pengimplementasian sistem dan analisis dari program yang telah dibuat. Bab 5 – Penutup yang berisi kesimpulan dari hasil analisis dan saran.

2. Studi Terkait

2.1. Penelitian Terkait

Terdapat beberapa penelitian terkait metode penerapan algoritma RSA dan SHA256 untuk keamanan sebuah dokumen yang dijadikan sebagai acuan dan bahan kajian pada penelitian ini.

Sihan Long pada penelitian [21] menganalisa 3 jenis algoritma *hasing* yaitu MD5, SHA-1 dan SHA512. Hasil dari penelitian tersebut adalah bahwa jika prioritas keamanan diutamakan maka bisa menggunakan SHA512, jika mempertimbangkan kecepatan dan efisiensi enkripsi maka dapat memilih MD5, dan jika mempertimbangkan kecepatan dan efisiensi keamanan maka SHA1 dapat dipilih.

M.A.Sadikin melakukan penelitian [22] memiliki hasil bahwa kedua algoritma ini dapat digunakan untuk enkripsi pada rekam kesehatan, dengan hasil *output black box* sesuai dengan yang diharapkan, sedangkan untuk pengujian *white box* menunjukan nilai *error density* yang bervariasi dengan proyek sekitar 16.000 baris kode.

Hassan Mansour melakukan analisis pada penelitian [23] menghasilkan bahwa walau lama waktu pembuatan kunci dipengaruhi oleh panjang kunci namun waktu yang dibutuhkan untuk membentuk kunci tidak terlalu signifikan sehingga tidak membebani sebuah sistem.

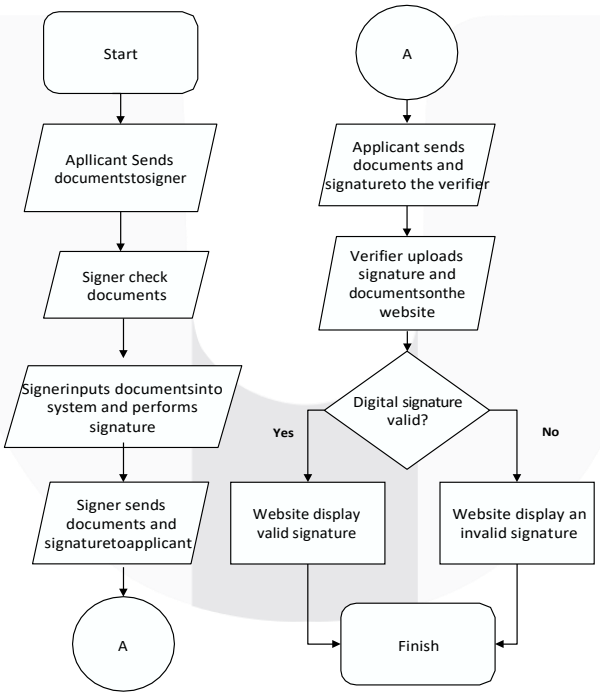
N. Jansma dan B. Arrendondo dalam penelitian [24] mendapati hasil bahwa RSA pembuatan kunci lebih lambat dari pada ECC untuk besar kunci 1024 bit diatasnya, namun proses verifikasi lebih cepat dibanding ECC.

E.C. Prabowo dan I. Afiranto pada penelitian [25] menghasilkan sebuah penelitian bahwa kombinasi RSA dan SHA 256 mampu memberikan layanan keamanan berupa otentikasi dokumen.

3. Sistem yang Dibangun

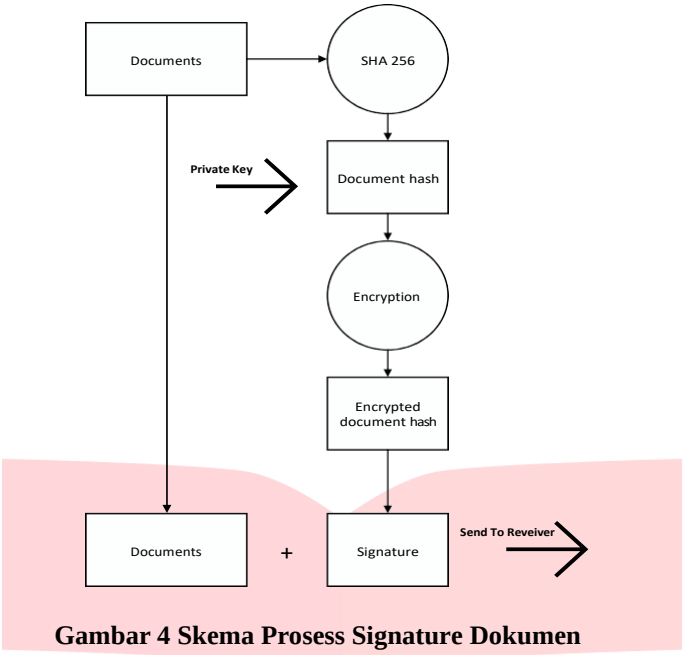
3.1. Sistem Model

Gambar .3 merupakan proses bisnis dari sistem yang dibangun adapun 3 aktor yang berperan dalam proses *digital signature* hingga selesai yaitu *signer*, *applicant*, dan *verifier*. Proses sistem dimulai oleh *applicant* yang melakukan *uploading* dokumen ke dalam sistem untuk mengirimkan dokumen kepada *signer* di mana tujuannya adalah meminta *signer* menandatangani dokumen *applicant*. Dokumen ini akan dikirimkan melalui perantara email, dan email yang digunakan pada sistem ini yaitu gmail. Setelah dokumen diterima, selanjutnya *signer* akan melakukan tandatangan dokumen dengan *uploading* dokumen ke dalam sistem dengan memilih kunci RSA sekaligus mengirimkan dokumen yang ditandatangani kepada *applicant* kembali. Kemudian *applicant* akan menerima balasan gmail yang berisikan dokumen dan *signature* yang dikirim oleh *signer*. *Applicant* akan meneruskan dokumen dan *signature* yang diperoleh kepada *verifier* melalui proses *uploading* pada aplikasi. Setelah *verifier* mendapatkan balasan email berupa dokumen dan *signature* selanjutnya *verifier* akan melakukan *uploading* dokumen beserta *signature* ke dalam aplikasi. Sistem akan memproses dan melakukan *output*, jika dokumen dan *signature* yang dimasukan adalah valid maka sistem akan menampilkan “valid signature”. Sebaliknya, jika terjadi pemalsuan terhadap salah satu *signature* atau dokumen atau keduannya maka sistem akan menampilkan “invalid signature” kemudian proses selesai.



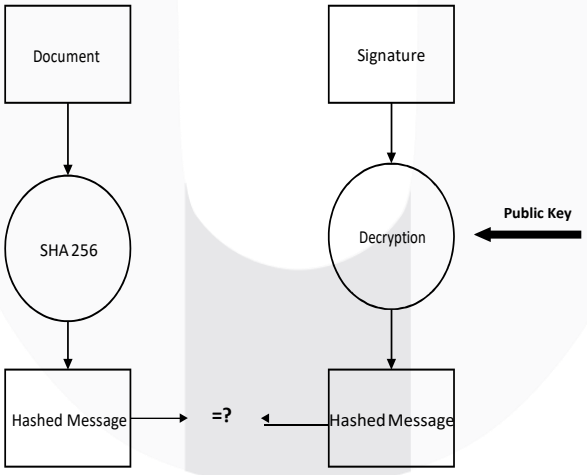
Gambar 3 Proses Bisnis Sistem

Gambar .4 merupakan skema proses *signature* yang hanya menggunakan dokumen untuk parameter proses *signature*, tahap pertama sistem akan melakukan hash menggunakan SHA256, hasil dari proses *hashing* akan menghasilkan *message digest*. nilai ini merupakan nilai yang mewakili integritas pada dokumen yang sedang diproses kemudian *message digest* akan dienkripsi menggunakan kunci privat pengirim, nilai ini disebut *signature*, kemudian dokumen dan *signature* akan dikirim ke penerima.



Gambar 4 Skema Prosess Signature Dokumen

Gambar .5 adalah skema proses verifikasi dimana aktor yang melakukan ini adalah verifikator. Parameter verifikasi membutuhkan 2 hal yaitu dokumen dan *signature*. Verifikator akan memasukkan 2 parameter tersebut ke dalam sistem. Sistem kemudian akan meng-hash dokumen untuk mendapatkan *message digest*. Sedangkan tanda tangan yang diperoleh akan didekripsi menggunakan kunci publik pengirim. Hasil dari dekripsi adalah *message digest*. Kemudian hasil dari kedua *message digest* yang didapat akan dibandingkan, apakah sama atau tidak. Jika *message digest* sama, maka dapat dikatakan otentik dan jika tidak, maka dokumen tersebut telah dipalsukan.



Gambar 5 Skema Proses Verifikasi Dokumen

3.2. Skenario Pengujian

Dua skenario tanda tangan digital dibuat untuk perbandingan, satu menggunakan enkripsi RSA 2048 bit dan yang lainnya menggunakan enkripsi RSA 4098 bit. Kode menunjukkan panjang kunci RSA dalam bit. Dari model sistem yang diusulkan, dilakukan simulasi untuk menentukan tiga hal. Yang pertama adalah analisis *overhead* untuk waktu yang diperlukan untuk pemrosesan *signature* dan verifikasi antara RSA 2048 bit dan RSA 4096 bit. Waktu proses dihitungan dengan lama waktu program dalam melakukan *signature* atau verifikasi dokumen. Hal kedua dan ketiga berturut-turut adalah analisis *confidentiality* dan analisis *integrity*. Adapun spesifikasi dokumen

yang digunakan pada penelitian ini yaitu, dokumen yang digunakan memiliki format PDF, dan ukuran dokumen yang diujikan memiliki variasi ukuran 0 hingga 10 Mb. Simulasi ini dilakukan sebanyak 30 kali dan diujikan pada aplikasi berbasis website *dummy* pada server lokal dengan spesifikasi CPU Intel® CoreTMi5-9500F@3.20GHz, RAM 8 GB, SSD 256 GB, GPU RX 580 dan Sistem Operasi 64-bit.

4. Evaluasi

4.1. Security Analysis : *Integrity*

Penggunaan tanda tangan digital adalah penandatangan dapat menjamin integritas dokumen. Ada dua ancaman integritas dokumen: yang pertama adalah dokumen diubah setelah tanda tangan digital dibuat, yang kedua adalah signature yang telah terbentuk kemudian dilakukan pemalsuan. Pada bagian ini, empat skenario dibuat yang dijabarkan dalam tabel 1. Pemalsuan yang dimaksud pada penelitian ini dengan merubah konten dari dokumen ataupun *signature* yang sudah dilakukan *siganturing*. Pada skenario pertama, dokumen dipalsukan setelah proses *signaturing* dan hasil *signature* yang terbentuk juga dipalsukan. Dalam skenario kedua, dokumen yang telah dilakukan *signaturing* kemudian dilakukan pemalsuan namun dengan hasil *signature* tetap asli. Dalam skenario ketiga, dokumen tidak dipalsukan setelah proses *signaturing*, tetapi *signature* yang telah dibuat dipalsukan. Dalam skenario keempat, dokumen tidak dipalsukan setelah proses *signaturing* dan *signature* tetap asli.

Tabel 1 Hasil Pengujian Parameter Keamanan *Integrity*

No.	Dokumen	Signature	Kondisi Pengujian
1.	Dipalsukan	Dipalsukan	*Invalid Signature*
2.	Dipalsukan	Tidak dipalsukan	*Invalid Signature*
3.	Tidak dipalsukan	Dipalsukan	*Invalid Signature*
4.	Tidak dipalsukan	Tidak dipalsukan	*Valid Signature*

Tabel 1 menunjukkan bahwa aspek *integrity* telah terjamin menggunakan metode yang diusulkan pada penelitian ini.

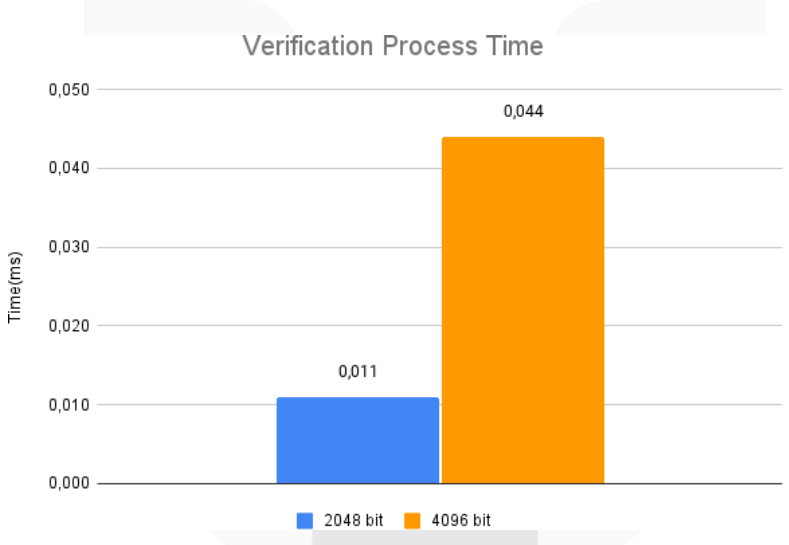
Melalui pengujian integritas, berbagai skenario ancaman dapat dideteksi, tetapi skenario seperti ini dapat dibayangkan: Seorang pemohon memalsukan dokumen, kemudian karena kurangnya akurasi, penandatangan memberikan tanda tangan digital. Dengan skenario ini, meskipun dokumen tersebut adalah dokumen palsu, verifikator akan tetap mendeteksinya sebagai tanda tangan yang valid. Metode tersebut adalah rekayasa sosial dan dalam studi lain dianalisis dalam metode keamanan lainnya [27].

4.2. Overhead Analysis

Bagian pertama dari skenario pengujian pertama adalah menguji *overhead* dari proses tanda tangan. Gambar 6 menunjukkan hasil dari waktu *overhead* yang dibutuhkan untuk membuat tanda tangan. Waktu yang ditunjukkan adalah waktu rata-rata yang diperoleh setelah melakukan percobaan sebanyak 30 kali. Hasil pengujian menunjukkan bahwa kunci RSA 4096 bit memiliki waktu yang lebih lama dibandingkan RSA 2048 bit dalam proses tanda tangan. Dimana selisih waktunya adalah 0,089 ms dan RSA 4096 bit kurang lebih membutuhkan waktu proses 4 kali lebih lama dari RSA 2048 bit.

Gambar 6 Performansi Proses *Signature*

Bagian kedua dari skenario pengujian pertama adalah pengujian *overhead* dari proses verifikasi. Gambar 7 menunjukkan hasil dari waktu *overhead* yang diperlukan untuk memverifikasi tanda tangan. Waktu yang ditunjukkan adalah waktu rata-rata yang diperoleh setelah melakukan percobaan sebanyak 30 kali. Hasil pengujian menunjukkan bahwa kunci RSA 4096 bit memiliki waktu yang lebih lama dibandingkan dengan RSA 2048 bit dalam proses verifikasi. Di mana perbedaan waktu 0,33 ms dan RSA 4096 bit membutuhkan waktu proses verifikasi 4 kali lebih lama dari RSA 2048 bit.



Gambar 7 Performansi Proses Verifikasi

Dari hasil pengujian *overhead* RSA 4096 bit membutuhkan waktu kurang lebih 4 kali dari RSA 2048 bit untuk proses *signature* dan proses verifikasi. Namun, waktu prosesnya masih lebih cepat daripada waktu pemrosesan untuk *paper* lain [16] [28], sehingga dengan tingkat keamanan yang ditawarkan seperti pada pengujian *brute force*, pengguna dapat memilih untuk menggunakan RSA 4096 bit.

4.3. Analisis *Brute Force Attack*

Dalam makalah ini, analisis *confidentiality* diuji dengan perhitungan simulasi serangan *brute force*. Lamanya waktu yang diperlukan untuk serangan *brute force* untuk memecahkan kunci tergantung pada berapa banyak kemungkinan kunci yang ada. Rumus untuk jumlah kunci yang mungkin adalah 2x, di mana x adalah panjang kunci dalam bit. Kemudian, jumlah upaya yang perlu dilakukan dalam serangan *brute force* adalah setengah dari jumlah kunci yang mungkin. Dalam ketahanan, diasumsikan bahwa dalam satu detik, peretas melakukan 10^6 percobaan per detik. Jadi, waktu yang dibutuhkan untuk memecahkan kunci dengan *brute force* adalah jumlah percobaan dibagi 10^6 x 1 detik. Jika hasilnya sangat besar, dapat dibagi dengan

jumlah detik per menit, jumlah menit per jam, jumlah jam per hari, dan seterusnya.

Pada tabel 2, waktu serangan *brute force* untuk RSA 2048 bit dan RSA 4096 bit dihitung dengan RSA 16 bit dan RSA 128 bit untuk perbandingan. Dari hasil waktu yang dibutuhkan untuk *brute force attack* paling lama adalah saat menggunakan RSA 4096 bit. Kunci RSA 4096 bit untuk *cracking* kunci membutuhkan 3,2 x 10^{616} kali lebih lama dari RSA 2048 bit.

Tabel 2 *Brute Force Attack*

RSA n (bits)	Banyaknya P	Number of Trials	Time required
16 [29]	65,536	32768	3,7 milliseconds
128 [29]	3,4 x 10^{38}	1.70 × 10^{38}	35.8 minutes
2048	3.23 × 10^{616}	1.61 × 10^{616}	5.1 × 10^{602} years
4096	1,04 x 10^{1233}	5.22 × 10^{1232}	1.65 x 10^{1219} years

5. Kesimpulan

Dalam makalah ini telah dilakukan kombinasi varian RSA 2048 bit atau RSA 4096 bit dengan SHA256 untuk diterapkan sebagai tanda tangan digital pada dokumen melalui aplikasi web untuk penggunaan perusahaan. Dari hasil pengujian *overhead* RSA 4096 bit membutuhkan waktu kurang lebih 4 kali dari waktu *overhead* RSA 2048 bit untuk proses *signature* dan proses verifikasi. Namun, melalui perhitungan simulasi *brute force*, RSA 4096 bit membutuhkan sekitar 10^{616} kali lebih lama untuk retak dibandingkan dengan RSA 2048 bit. Selain itu, melalui uji integritas, *verifier* dapat mendeteksi jika dokumen atau tanda tangan dipalsukan. Adapun saran untuk penelitian selanjutnya dapat melakukan analisis rekayasa sosial pada dokumen tanda tangan digital dan menggunakan *Mean Opinion Score* untuk mengukur kualitas aplikasi yang dibangun.

Referensi

[1] E. Mahase, "Covid-19: WHO declares pandemic because of 'alarming levels' of spread, severity, and inaction," *BMJ*, p. m1036, Mar. 2020, doi: 10.1136/bmj.m1036.

[2] T. Singhal, "A Review of Coronavirus Disease-2019 (COVID-19)," *Indian J. Pediatr.*, vol. 87, no. 4, pp. 281–286, Apr. 2020, doi: 10.1007/s12098-020-03263-6.

[3] KOMINFO, "Berlaku Mulai 22 Juni, Inilah Ketentuan Pengetatan PPKM Mikro." https://kominfo.go.id/content/detail/35155/berlaku-mulai-22-juni-inilah-ketentuan-pengetatan-ppkm-mikro/0/berita (accessed Jul. 13, 2021).

[4] KOMINFO, "Penggunaan Internet Naik 40% Saat Bekerja dan Belajar dari Rumah." https://www.kominfo.go.id/content/detail/25881/penggunaan-internet-naik-40-saat-bekerja-dan-belajar-dari-rumah/0/berita_satker (accessed Jun. 10, 2021).

[5] Arpit Jain Singhai and D. Faizan, "Standard Operational Mechanism for E-Govemance during Pandemics: An Indian Case Study," in *2021 8th International Conference on Computing for Sustainable Global Development (INDIACom)*, 2021, pp. 250–254.

[6] Z. Li and X. Liu, "An examination of handwritten signatures forged using photosensitive signature stamp," *Forensic Sci. Res.*, vol. 6, no. 2, pp. 168–182, Apr. 2021, doi: 10.1080/20961790.2021.1898755.

[7] L. Wang, C. Huang, and H. Cheng, "Quantum attack-resistant signature scheme from lattice cryptography for WFH," in *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, 2021, pp. 868–871.

[8] A. Shatnawi, E. V. Munson, and C. Thao, "Maintaining Integrity and Non-Repudiation in Secure Offline Documents," in *Proceedings of the 2017 ACM Symposium on Document Engineering*, Valletta Malta, Aug. 2017, pp. 59–62. doi: 10.1145/3103010.3121038.

[9] R. A. D. Jesus, "Challenges and Opportunities in the Field Of Information and Communications Technology (Ict) due to Covid-19 Pandemic and Migration Towards The New Normal," no. 11, p. 8, 2021.

[10] W. Stallings, *Cryptography and network security: principles and practice*, Seventh edition. Boston: Pearson, 2017.

[11] R. Munir, "Pengantar Kriptografi," p. 68.

[12] R. Munir, "PENGGUNAAN TANDA-TANGAN DIGITAL UNTUK MENJAGA INTEGRITAS BERKAS PERANGKAT LUNAK," p. 4, 2005.

[13] M. Ihwani, "MODEL KEAMANAN INFORMASI BERBASIS DIGITAL SIGNATURE DENGAN ALGORITMA RSA," vol. 1, no. 1, p. 6, 2016.
[14] Pooja and M. Yadav, "Digital Signature," *Int. J. Sci. Res. Comput. Sci. Eng. Andin. Technol.*, vol. 3, no. 6, 2018.
[15] alireza shabani, "A review of digital signature Schemes, Based on Asymmetric cryptography andtheir applications,." Iran. J. Mar. Sci. Technol, 2020.
[16] F. J. Aufa, Endroyono, and A. Affandi, "Security System Analysis in Combination Method: RSA Encryption and Digital Signature Algorithm," in *2018 4th International Conference on Science and Technology (ICST)*, Yogyakarta, Aug. 2018, pp. 1–5. doi: 10.1109/ICSTC.2018.8528584.
[17] N. Y. Goshwe, "Data Encryption and Decryption Using RSA Algorithm in a Network Environment," p. 5, 2013.
[18] F. Ahmad and L.-M. Cheng, "Paper Document Authentication Using Print-Scan Resistant Image Hashing and Public-Key Cryptography," in *Security, Privacy, and Anonymity in Computation, Communication, and Storage*, vol. 11611, G. Wang, J. Feng, M. Z. A. Bhuiyan, and R. Lu, Eds. Cham: Springer International Publishing, 2019, pp. 157–165. doi: 10.1007/978-3-030-24907-6_13.
[19] F. Jahan, M. Mostafa, and S. Chowdhury, "SHA-256 in Parallel Blockchain Technology: Storing Land Related Documents," *Int. J. Comput. Appl.*, vol. 175, no. 35, pp. 33–38, Dec. 2020, doi: 10.5120/ijca2020920911.
[20] D. J. Prathyusha and K. Govinda, "Securing virtual machines from DDoS attacks using hash-based detection techniques," *Multiagent Grid Syst.*, vol. 15, no. 2, pp. 121–135, Jul. 2019, doi: 10.3233/MGS-190305.
[21] S. Long, "A Comparative Analysis of the Application of Hashing Encryption Algorithms for MD5, SHA-1, and SHA-512," *J. Phys. Conf. Ser.*, vol. 1314, p. 012210, Oct. 2019, doi: 10.1088/1742-6596/1314/1/012210.
[22] M. A. Sadikin and R. W. Wardhani, "IMPLEMENTATION OF RSA 2048-BIT AND AES 256-BIT WITH DIGITAL SIGNATURE FOR SECURE ELECTRONIC HEALTH RECORD APPLICATION," p. 7.
[23] A. H. Mansour, "Analysis of RSA Digital Signature Key Generation using Strong Prime," vol. 24, no. 1, p. 9, 2017.
[24] N. Jansma and B. Arrendondo, "Performance Comparison of Elliptic Curve and RSA Digital Signatures," p. 20.
[25] E. C. Prabowo and I. Afrianto, "PENERAPAN DIGITAL SIGNATURE DAN KRIPTOGRAFI PADA OTENTIKASI SERTIFIKAT TANAH DIGITAL," *Komputa J. Ilm. Komput. Dan Inform.*, vol. 6, no. 2, pp. 83–90, Oct. 2017, doi: 10.34010/komputa.v6i2.2481.
[26] WIKIPEDIA, "SHA-2." https://en.wikipedia.org/wiki/SHA-2 (accessed Aug. 02, 2021).
[27] P. L. Gallegos-Segovia, J. F. Bravo-Torres, V. M. Larios-Rosillo, P. E. Vintimilla-Tapia, I. F. Yuquilima-Albarado, and J. D. Jara-Saltos, "Social engineering as an attack vector for ransomware," in *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, Pucon, Oct. 2017, pp. 1–6. doi: 10.1109/CHILECON.2017.8229528.W.
[28] A. I. Ali, "Comparison and Evaluation of Digital Signature Schemes Employed in NDN Network," *Int. J. Embed. Syst. Appl.*, vol. 5, no. 2, pp. 15–29, Jun. 2015, doi: 10.5121/ijesa.2015.5202.
[29] W. Stallings, *Data and computer communications*, 8th ed. Upper Saddle River, N.J: Pearson/Prentice Hall, 2007.