PERGURUAN TINGGI MUHAMMADIYAH
UNIVERSITAS AHMAD DAHLAN
LEMBAGA PENELITIAN DAN PENGABDIAN KEPADA MASYARAKAT
UAD Kampus 2 Unit B, Jl. Pramuka No. 5F, Pandeyan, Umbulharjo, Yogyakarta 55161, email : lppm@uad.ac.id

SURAT PERJANJIAN PELAKSANAAN PENELITIAN
Nomor : PTM-277/SP3/LPPM-UAD/VI/2021

Pada hari ini, Selasa tanggal Satu bulan Juni tahun Dua ribu dua puluh satu (01-06-2021), kami yang bertandatangan di bawah ini :

1. Nama : Anton Yudhana, S.T., M.T., Ph.D.
   Jabatan : Kepala Lembaga Penelitian dan Pengabdian Kepada Masyarakat Universitas Ahmad Dahlan (LPPM UAD), selanjutnya disebut sebagai PIHAK PERTAMA.
2. Nama : Dr. Imam Riadi, M.Kom
   Jabatan : Dosen/Peneliti pada Program Studi Sistem Informasi Fakultas Sains dan Teknologi Terapan (FAST) Universitas Ahmad Dahlan (UAD), selaku Ketua Peneliti, selanjutnya disebut PIHAK KEDUA.

Kedua belah pihak menyatakan setuju dan mufakat untuk mengadakan perjanjian pelaksanaan penelitian untuk selanjutnya disebut Surat Perjanjian Pelaksanaan Penelitian (SP3) dengan ketentuan dan syarat-syarat sebagai berikut.

Pasal 1
DASAR HUKUM

(1) Hasil review/penilaian proposal yang dilakukan oleh Tim Reviewer Penelitian Internal UAD.

(2) Surat Keputusan Kepala LPPM UAD Nomor: U12.3/316/V/2021 tanggal 29 Mei 2021 tentang Penetapan Hasil Seleksi Proposal Penelitian Dana UAD Tahun Akademik 2020/2021.

Pasal 2
RUANG LINGKUP DAN JANGKA WAKTU PENELITIAN

(1) PIHAK PERTAMA memberikan pekerjaan kepada PIHAK KEDUA dan PIHAK KEDUA menyatakan menerima pekerjaan dari PIHAK PERTAMA berupa kegiatan penelitian sebagai berikut :

Skema	:	Penelitian Tesis Magister
Judul penelitian	:	Optimalisasi Layanan Autentikasi Wireless IEEE 802.1x Menggunakan Teknologi Blockchain
Jenis Riset	:	Dasar, TKT : 3
Luaran Wajib	:	Artikel di jurnal Internasional, Artikel di Jurnal Nasional Terakreditasi

(2) Jangka waktu pelaksanaan penelitian tersebut pada ayat (1) dimulai sejak ditandatangani SP3 ini sampai dengan batas akhir unggah Laporan Akhir Penelitian pada tanggal 31 Desember 2021

Pasal 3

PERSONALIA PELAKSANA PENELITIAN

Personalia pelaksana penelitian ini terdiri dari :

Ketua Peneliti	:	Dr. Imam Riadi, M.Kom
Pembimbing	:	-
Anggota	:	

Pasal 4

BIAYA PENELITIAN DAN CARA PEMBAYARAN

(1) PIHAK PERTAMA menyediakan dana pelaksanaan penelitian kepada PIHAK KEDUA sejumlah Rp. 11.700.000,00 (Sebelas Juta Tujuh Ratus Ribu Rupiah) yang dibebankan pada Anggaran Pendapatan dan Belanja (APB) LPPM UAD Tahun Akademik 2020/2021 dibayarkan melalui rekening bank atas nama Ketua Peneliti oleh Biro Keuangan dan Anggaran UAD sebagai berikut :

Nama	:	Dr. Imam Riadi, M.Kom
Nama Bank	:	BPD DIY SYARIAH
Nomor Rekening	:	801211007878

(2) Tahap I sebesar 70% x Rp 11.700.000,00 = Rp 8.190.000,00 (delapan juta seratus sembilan puluh ribu Rupiah), dibayarkan setelah SP3 ini ditandatangani oleh PARA PIHAK dan PIHAK KEDUA telah mengunggah file kontrak SP3 ini pada portal Penelitian UAD.

(3) Tahap II sebesar 30% x Rp 11.700.000,00 = Rp 3.510.000,00 (tiga juta lima ratus sepuluh ribu Rupiah), dibayarkan setelah (a) PIHAK KEDUA mengunggah Laporan Akhir Penelitian dan (b) luaran wajib penelitian dinyatakan tercapai.

(4) Jika sampai pada batas akhir penelitian PIHAK KEDUA hanya dapat mengunggah Laporan Akhir Penelitian dan TIDAK DAPAT merealisasikan luaran wajib, maka dana penelitian Tahap II hanya dapat dicairkan sebesar 15%.

Pasal 5

PELAKSANAAN PEMBIMBINGAN

(1) Khusus skema Penelitian Dosen Pemula (PDP), peneliti wajib melakukan pembimbingan atau konsultasi dengan dosen pembimbing penelitian paling sedikit 4 (empat) kali pembimbingan.

(2) Pembimbingan sebagaimana dimaksud dalam ayat (1) antara lain dalam hal-hal berikut.
a. penyusunan angket/kuesioner dan atau teknik pengumpulan data lainnya;
b. analisis data dan interpretasinya;
c. penyusunan hasil penelitian, pembahasan, penarikan kesimpulan;
d. penyusunan luaran penelitian.

(3) Pembimbingan sebagaimana dimaksud dalam ayat (1) dan ayat (2) dituliskan sesuai dengan template form pembimbingan yang tersedia.

Pasal 6

JENIS LAPORAN PENELITIAN

(1) PIHAK KEDUA wajib menyusun dan mengungah laporan penelitian melalui portal Penelitian UAD yang terdiri atas :
   a. Laporan Kemajuan
   b. Laporan Akhir
(2) Berkas Laporan Kemajuan digunakan sebagai bahan monitoring dan evaluasi (monev) internal, diunggah selambat-lambatnya tanggal 15 Oktober 2021.
(3) Berkas Laporan Akhir digunakan sebagai acuan pencairan dana Tahap II dan bahan pertimbangan berlanjut atau tidaknya kontrak penelitian tahun jamak (multi years), diunggah selambat-lambatnya tanggal 31 Desember 2021.

Pasal 7

LUARAN WAJIB PENELITIAN

(1) PIHAK PERTAMA berkewajiban untuk merealiasikan luaran wajib penelitian sebagaimana yang dijanjikan dalam proposal.
(2) Status minimal luaran wajib yang harus dicapai oleh PIHAK KEDUA adalah sebagai berikut. (i) accepted untuk jenis luaran artikel jurnal/seminar/konferensi, atau (ii) naik cetak untuk jenis luaran buku, atau (iii) diterima atau dibahas instansi pengguna untuk jenis luaran naskah akademik, atau (iv) telah terdaftar atau didaftarkan untuk jenis kekayaan intelektual (KI), atau (v) telah terwujud atau telah dilakukan uji laboratorium untuk jenis luaran purwarupa (prototipe), dan sejenisnya.

Pasal 8

MONITORING DAN EVALUASI

(1) PIHAK PERTAMA berhak untuk melakukan monitoring dan evaluasi (monev) pelaksanaan penelitian, baik secara administrasi maupun substansi.
(2) Pemantauan kemajuan penelitian dilakukan oleh Tim Monev yang dibentuk oleh PIHAK PERTAMA.
(3) Monev internal dilakukan terhadap dokumen Laporan Kemajuan yang diunggah oleh PIHAK KEDUA.
(4) PIHAK PERTAMA berhak untuk menentukan lanjut atau putusnya kontrak penelitian tahun jamak (multi years) berdasarkan hasil dari monev tahap II terhadap Laporan Akhir dan capaian luaran penelitian tahun berjalan yang diunggah PIHAK KEDUA.

Pasal 9

TANGGUNGAN PENELITIAN DAN LUARAN PENELITIAN

(1) Peneliti dinyatakan memiliki tanggungan penelitian apabila sampai pada masa penerimaan proposal penelitian periode berikutnya belum menyelesaikan kewajiban unggah Laporan Akhir Penelitian.
(2) Peneliti yang memiliki tanggungan penelitian sebagaimana dimaksud pada ayat (1) tidak diperkenankan mengajukan proposal penelitian pada periode tersebut.
(3) Peneliti dinyatakan memiliki tanggungan luaran penelitian apabila sampai pada masa akhir unggah Laporan Akhir Penelitian, luaran wajib belum tercapai dengan status minimal seperti disebutkan pada Pasal 7 ayat (2).

(4) Peneliti yang memiliki tanggungan luaran penelitian sebagaimana dimaksud pada ayat (3) masih diperkenankan mengajukan proposal penelitian pada periode terdekat.

(5) Peneliti yang belum memenuhi luaran wajib sampai pada penerimaan proposal penelitian pada periode tahun berkutnya tidak diperkenankan mengajukan proposal pada periode tersebut.

(6) Tanggungan penelitian dan/atau luaran wajib penelitian berlaku bagi Ketua dan Anggota peneliti dari Universitas Ahmad Dahlan.

Pasal 10
SANKSI DAN PEMUTUSAN PERJANJIAN PENELITIAN

(1) PIHAK PERTAMA berhak memberikan peringatan dan atau teguran atas kelalaian dan atau pelanggaran yang dilakukan oleh PIHAK KEDUA yang mengakibatkan tidak dapat terpenuhinya kontrak penelitian ini.

(2) PIHAK PERTAMA berhak melakukan pemutusan perjanjian penelitian, jika PIHAK KEDUA tidak mengindahkan peringatan yang diberikan oleh PIHAK PERTAMA.

(3) Segala kerugian material maupun finansial yang disebabkan akibat kelalaian PIHAK KEDUA, maka sepenuhnya menjadi tanggungjawab PIHAK KEDUA.

(4) Jenis sanksi yang diberikan dapat berupa :
(a) tidak diperkenankannya mengajukan proposal penelitian sebagaimana dimaksud pada Pasal 9 ayat (5) sampai kewajibannya terselesaikan; dan atau
(b) tidak dapat mencairkan dana Tahap II; dan atau
(c) mengembalikan dana yang telah diterima oleh PIHAK KEDUA.

Pasal 11
KEADAAN MEMAKSA (FORCE MAJEUR)

Ketentuan dalam Pasal 10 tersebut di atas tidak berlaku dalam keadaan sebagai berikut :

a. Keadaan memaksa (force majeur)
b. PIHAK PERTAMA menyetujui atas terjadinya keterlambatan yang didasarkan pada pemberitahuan sebelumnya oleh PIHAK KEDUA kepada PIHAK PERTAMA dengan Surat Pemberitahuan mengenai kemungkinan terjadinya keterlambatan dalam penyelesaian kegiatan penelitian sebagaimana dimaksud dalam Pasal 2; dan sebaliknya PIHAK KEDUA menyetujui terjadinya keterlambatan pembayaran sebagai akibat keterlambatan dalam penyelesaian perjanjian penelitian.

Pasal 12

(1) Keadaan memaksa (force majeur) sebagaimana yang dimaksud dalam Pasal 11 ayat (1) adalah peristiwa-peristiwa yang secara langsung mempengaruhi pelaksanaan perjanjian serta terjadi di luar kekuasaan dan kemampuan PIHAK KEDUA ataupun PIHAK PERTAMA.

(2) Peristiwa yang tergolong dalam keadaan memaksa (force majeur) antara lain berupa bencana alam, pemogokan, wabah penyakit, huru-hara, pemberontakan, perang, waktu kerja diperpendek oleh pemerintah, kebakaran dan atau peraturan pemerintah mengenai keadaan bahaya serta hal-hal lainnya yang dipersamakan dengan itu, sehingga PIHAK KEDUA ataupun PIHAK PERTAMA terpaksa tidak dapat memenuhi kewajibannya.

(3) Peristiwa sebagaimana dimaksud pada ayat (2) tersebut di atas, wajib dibenarkan oleh penguasa setempat dan diberitahukan dengan surat pemberitahuan oleh PIHAK KEDUA kepada PIHAK PERTAMA atau PIHAK PERTAMA kepada PIHAK KEDUA yang menyebutkan telah terjadinya peristiwa yang dikategorikan sebagai keadaan memaksa (force majeur).

(4) PIHAK PERTAMA memberikan kesempatan kepada PIHAK KEDUA untuk menyelesaikan perjanjian kontrak ini sampai pada batas waktu yang disepakati oleh PARA PIHAK jika keadaan force majeur dinyatakan telah selesai.

Pasal 13
PENYELESAIAN PERSELISIHAN

(1) Apabila dalam pelaksanaan perjanjian dan segala akibatnya timbul perbedaan pendapat atau perselisihan, PIHAK PERTAMA dan PIHAK KEDUA setuju untuk menyelesaikannya secara musyawarah untuk mencapai mufakat.

(2) Apabila penyelesaian sebagaimana termaksud dalam ayat (1) di atas tidak tercapai, maka PIHAK PERTAMA dan PIHAK KEDUA sepakat menyerahkan perselisihan tersebut melalui mediasi dengan Rektor sebagai atasan langsung dari PIHAK PERTAMA yang putusannya bersifat final dan mengikat.

Pasal 14
PENGUNDURAN DIRI

(1) Apabila PIHAK KEDUA mengundurkan diri atau membatalkan SP3 ini, maka PIHAK KEDUA wajib mengajukan Surat Pengunduran Diri yang ditujukan kepada PIHAK PERTAMA.

(2) Surat Pengunduran Diri sebagaimana dimaksud pada ayat (1) wajib disahkan oleh dekan fakultas ketua peneliti yang bersangkutan.

(3) PIHAK KEDUA wajib mengembalikan dana yang telah diterima kepada PIHAK PERTAMA

Pasal 15
LAIN-LAIN

(1) Hal-hal yang dianggap belum cukup dan perubahan-perubahan perjanjian akan diatur kemudian atas dasar permufakatan kedua belah pihak yang akan dituangkan dalam bentuk Surat atau Perjanjian Tambahan (addendum), yang merupakan satu kesatuan dan bagian yang tidak terpisahkan dari perjanjian awal.

(2) Surat Perjanjian Pelaksanaan Penelitian (SP3) ini berlaku sejak ditandatangani dan disetujui oleh PARA PIHAK.

PIHAK PERTAMA,

Anton Yudhana, S.T., M.T., Ph.D.
NIP/NIY. 60010383

PIHAK KEDUA,

Dr. Imam Riadi M.Kom
NIP/NIY. 60020397

Rumpun Ilmu	:	Teknik Informatika
Bidang Kepakaran	:	Communications Technologies
Jenis Riset	:	Dasar

PROPOSAL PENELITIAN
SKEMA PENELITIAN TESIS MAGISTER

OPTIMALISASI LAYANAN AUTENTIKASI WIRELESS IEEE 802.1X MENGGUNAKAN TEKNOLOGI BLOCKCHAIN

TIM PENELITI :

Ketua : Dr. Imam Riadi, M.Kom

Mahasiswa Terlibat : 1. Aulyah Zakilah Ifani (1500018176)

SISTEM INFORMASI
SAINS DAN TEKNOLOGI TERAPAN
UNIVERSITAS AHMAD DAHLAN

NOVEMBER 2021

HALAMAN PENGESAHAN
LAPORAN AKHIR PENELITIAN DANA INTERNAL UAD
TAHUN AKADEMIK 2021/2022

Judul Penelitian	:	Optimalisasi Layanan Autentikasi Wireless IEEE 802.1x Menggunakan Teknologi Blockchain
Butir Renstra Prodi/Pusat	:	Program Studi
TSE Penelitian	:	10.01-Computer software and services
Jenis Riset	:	Dasar
Skala TKT	:	3

Ketua Peneliti

a. Nama Lengkap dan Gelar	:	Dr. IMAM RIADI M.Kom
b. NIY/NIP	:	60020397
c. Fakultas/Program Studi	:	Sains dan Teknologi Terapan / Sistem Informasi
d. Pendidikan Terakhir	:	S3
e. Jabatan Akademik	:	Lektor Kepala

Anggota Peneliti

Nama Lengkap dan Gelar	:	

Anggota Peneliti Eksternal

Nama Lengkap dan Gelar	:	

Jumlah mahasiswa terlibat	:	1 orang
Lama Penelitian	:	7 bulan
Biaya Total Penelitian	:	Rp. 11.700.000,00
- Dana Disetujui	:	Rp. 11.700.000,00
- Sumber Dana Lain	:	Rp. 0,00

Menyetujui,
Kepala LPPM Universitas Ahmad Dahlan,

Anton Yudhana, S.T., M.T., Ph.D.
NIP/NIY. 60010383

Yogyakarta, 06 Januari 2022
Ketua Pengusul,

Dr. IMAM RIADI M.Kom
NIP/NIY. 60020397

LAPORAN AKHIR PENELITIAN

Ringkasan penelitian berisi: (i) latar belakang penelitian, (ii) tujuan penelitian, (iii) tahapan metode penelitian, (iv) luaran yang ditargetkan, (v) uraian TKT penelitian yang ditargetkan serta (vi) hasil penelitian yang diperoleh sesuai dengan tahun pelaksanaan penelitian.

RINGKASAN

Keamanan data dan informasi pada suatu sistem menjadi hal yang sangat penting bagi pengguna. Seiring berkembangnya jaman, salah satu yang harus diperhatikan keamanannya yaitu sistem login yang membutuhkan autentikasi dalam keamanannya dengan bukti identitas atau kepemilikan. Dalam banyak kasus, informasi login pengguna disimpan di server sehingga memberikan akses ke informasi sensitif dan peretas dengan mudah membobol data dari pengguna.

Penelitian ini berfokus pada autentikasi keamanan data berupa username dan password pada sistem login. Autentikasi digunakan untuk menurunkan akses berbahaya dan meningkatkan keamanan proses autentikasi. Salah satu teknologi inovasi yang mampu menyelesaikan permasalahan tersebut adalah teknologi blockchain. Data atau transaksi dalam blockchain disimpan dalam bentuk hash sehingga menyulitkan peretas untuk membobolnya. Implementasi blockhain menggunakan bahasa pemrograman solidity untuk membangun smart contract, tools lain yang digunakan yaitu metamask, ganache, dan truffle. *Network Forensics Development Life Cycle* (NFLDC) digunakan sebagai framework dengan lima tahapan, yaitu *Initiation, Acquisition, Implementation, Operation*, dan *Disposition*. Pengujian dalam penelitian ini menggunakan *tools* wireshark dan networkminer.

Berdasarkan penelitian yang dilakukan, pengujian dilakukan sebanyak lima percobaan dengan dua skenario yaitu autentikasi sistem login sebelum menggunakan blockchain dan setelah menggunakan blockchain. Sistem yang dibangun menggunakan blockchain mampu mengamankan data, berdasarkan log data yang dihasilkan dari pengujian setelah diterapkan blockchain didapatkan hasil 97,8% yang menunjukkan bahwa tingkat keamanan sistem relatif tinggi dibandingkan dengan sistem sebelum diterapkan blockchain yang memiliki tingkat keamanan 45,1%.

Kata kunci maksimal 5 kata kunci. Gunakan tanda baca titik koma (;) sebagai pemisah dan ditulis sesuai urutan abjad.

Kata kunci : Autentikasi; *Blockchain*; *Networkminer*; NFLDC; *Wireshark*

Hasil pelaksanaan penelitian berisi: (i) kemajuan pelaksanaan penelitian yang telah dicapai sesuai tahun pelaksanaan penelitian, (ii) data yang diperoleh, (iii) hasil analisis data yang telah dilakukan, (iv) pembahasan hasil penelitian, serta (v) luaran yang telah didapatkan. Seluruh hasil atau capaian yang dilaporkan harus berkaitan dengan tahapan pelaksanaan penelitian sebagaimana direncanakan pada proposal. **Penyajian data** dan **hasil penelitian** dapat berupa gambar, tabel, grafik, dan sejenisnya, serta **pembahasan hasil penelitian** didukung dengan sumber pustaka primer yang relevan dan terkini.

LATAR BELAKANG

Zaman sekarang internet menjadi sangat terkenal, penggunaan internet bukan hanya di kalangan muda saja akan tetapi di kalangan dewasa pun banyak yang menggunakan internet sebagai kebutuhan. Kebutuhan internet ini yang menjadikan seseorang sangat mudah untuk mengaksesnya (Kunang, 2013). Wifi mudah di akses oleh pengguna hanya dengan beberapa perangkat contohnya seperti *smartphone*, laptop, dll (Rusdan, 2020). Tantangan terbesar dalam melindungi jaringan *wireless* dalam hal mobilitas pengguna yaitu serangan ilegal untuk mendapatkan *username* dan *password* dari pengguna (Rusdan, 2020). Terdapat beberapa riset yang telah membahas berbagai sistem pengamanan jaringan *wireless*, mulai dari pengamanan *access point* yang menerapkan MAC *address Filter*, *Service Set Identifier* (SSID), *Wired Equivalent Privacy* (WEP), *Wi-Fi Protected Access* (WPA), *Extensible Authentication Protocol* (EAP), dan juga sistem yang menggunakan autentikasi *captive portal.*

Kata sandi (password) adalah metode autentikasi yang paling sering digunakan dalam sistem keamanan. Kemudahan dalam implementasi menjadi alasan utama menggunakan sistem berbasis password (Rusdan, 2020). Autentikasi dengan username dan password sangat rentan terhadap serangan, berupa pencurian *username* dan *password*. Data tersebut sangat mudah diperoleh secara ilegal. Banyak pengguna yang tidak paham terhadap serangan yang terjadi, proses *login* tidak aman saat menggunakan perangkat, misalnya jaringan yang ada di sekolah, hotel, atau ditempat umum lainnya. Kata sandi (*password*) adalah metode autentikasi yang paling sering digunakan di berbagai sistem keamanan. Kemudahan dalam implementasi menjadi alasan utama menggunakan sistem berbasis *password* (Rusdan, 2020). Teknologi *blockchain* menjadi salah satu yang dapat menyelesaikan permasalahan tersebut. Sistem yang di bangun dengan menggunakan teknologi *blockchain* membuat peretas sulit mendapatkan *username* dan *password* pengguna apalagi dengan mengubah atau memodifikasi data. Hal itu dikarenakan data yang disimpan dimiliki setiap orang dan data harus sama sehingga sulit untuk memecahkan kode di setiap blok.

Nakamoto membuat protokol bitcoin menggunakan teknologi *Blockchain*. *Bitcoin* merupakan sebuah aplikasi atau penerapan dari *Blockchain*. *Blockchain* berisi transaksi-transaksi yang disimpan dalam sebuah blok data (Damai, 2019). Setiap blok memiliki nilai *hash* dan setiap blok mendapatkan nilai *hash* dari blok sebelumnya (Abdillah, 2020). Rantai blok yang baru saling tertaut antara satu blok dengan blok lainnya. Blok awal disebut blok genesis yang merupakan spesial blok yang dinomori nol dan diberi kode dalam aplikasi *blockchain* (Suyash Gupta, 2020). Keuntungan teknlogi *blockchain* adalah pada layanan pemantauan, kerahasiaan privasi integritas,

keamanan autentikasi. *Blockchain* memberikan keamanan dan consensus yang terdistribusi dan juga terbukti aman (Salman, 2019).

Blockchain memberi pengguna internet kemampuan untuk mengautentikasi informasi digital. Objek yang dapat digunakan dalam proses autentikasi dengan blockchain yaitu sistem login, smart payment (Ismanto, 2019), *crowfunding* (Harahap, 2019), dll. Proses autentikasi yang tidak diamankan dengan baik rentan terhadap pencurian data dari pengguna yang tidak bertanggung jawab. Pemanfaatan blockchain digunakan untuk mengamankan data-data dari pengguna. Seluruh data disimpan pada jaringan blockchain ke pemilik jaringan blockchain berdasarkan kesepakatan (consensus). Sulit bagi hacker untuk mengubah dan memodifikasi data yang sama di semua komputer pada saat yang bersamaan karena butuh waktu lama untuk memecahkan kode enkripsi blok data di jaringan komputer (Nurfaizi, 2019).

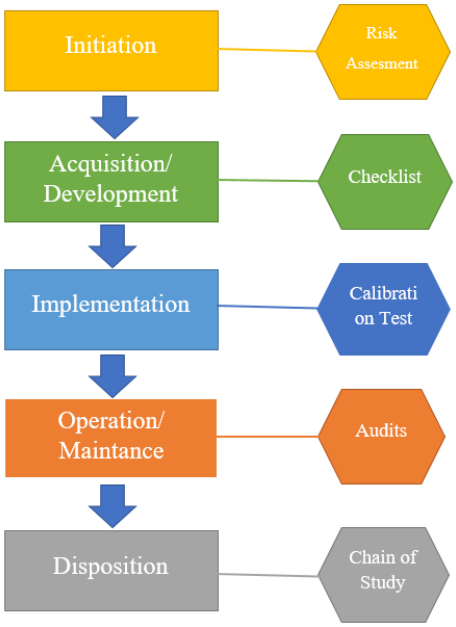
Objek yang digunakan dalam penelitian ini yaitu sistem login. Sistem login digunakan karena merupakan hal yang sangat perlu diperhatikan keamanannya. Data berupa username dan password sangat mudah diretas oleh orang-orang yang tidak bertanggung jawab, contohnya menggunakan tools wireshark, blackcbox testing (Pallavi, 2021), *networkminer* (Susianto, 2018), *burpsuite* (Riadi, 2020), *OWASP* (Riadi, 2020), dll. Hal tersebut yang menjadi dasar data dari pengguna harus diamankan. Sistem login membutuhkan autentikasi untuk mengamankan identitas dari pengguna, sehingga dalam penelitian ini menggunakan teknologi blockchain untuk mengamankan data dari pengguna dan framework yang digunakan dalam mendukung pengembangan sistem dan mengatasi kelemahan sistem yaitu Network Forensic Development Life Cycle (NFDLC).

NFDLC merupakan siklus yang diimplementasikan dalam pembuatan atau pengembangan sistem forensik jaringan yang berkualitas, mempermudah mendapatkan data, mengakuisasi data (Dhammearatchi, 2015). NFDLC mempunyai tahapan penelitian yang mudah dalam mengimplementasikannya serta memiliki integrasi tentang temuan yang diperoleh pada proses forensik. Tahapan dalam NFDLC yaitu Initiation, Acquisition, Implementation, Operation, dan Disposition. Framework yang juga dapat digunakan dalam beberapa penelitian yaitu Association of Chief Police Officers (ACPO) (Riadi, 2021), National Institute of Standards and Technology (NIST) (Umar, 2018), National of Justice (NIJ) (Riadi, 2018).

Penelitian ini bertujuan untuk mengamankan sistem keamanan login dengan autentikasi blockchain dan melakukan pengujian sistem. Sistem keamanan login di modifikasi dengan menggunakan teknologi *Blockchain*.

METODE PENELITIAN

Tahapan penelitian sebagai alur proses dari penelitian ini berpedoman pada framework *Network Forensic Development Life Cycle* (NFDLC). *Framework Network Forensic Development Life Cycle* (NFDLC) merupakan siklus yang diimplementasikan dalam pembuatan atau pengembangan sistem forensik jaringan yang berkualitas, mempermudah mendapatkan data, dan mengakuisisi data. Tahapan penelitian NFLDC berisikan *initiation, acquisition, implementation, operations dan disposition.* Gambar 1 menunjukkan *framework Network Forensic Development Life Cycle* (NFDLC).



Gambar 1. Metode Penelitian menggunakan *Network Forensic Development Life Cycle*

Gambar 1 menunjukkan metode NFLDC yang merupakan gabungan dari metode NFR dan ISDLC. Metode ini dipilih karena mempunyai tahapan penelitian yang praktis dan mudah untuk diimplementasikan serta memiliki integrasi tentang temuan yang diperoleh pada proses forensik. Tahapan-tahapan NFLDC sebagai berikut.

1. *Initiation*

   Tahap ini disebut tahap persiapan. Tahap mempersiapkan kebutuhan dari penelitian, mulai dari perangkat yang digunakan yang mencakup *hardware*, *software.* Tahapan ini merupakan tahapan menganalisis kebutuhan dari sistem.

2. *Acquisition*

   Tahap *acquisition* merupakan tahapan merancang pengembangan sistem dan perancangan *flowchart*. Perancangan dalam hal ini digunakan untuk mendukung pembuatan sistem. Tahapan rancangan sistem mengalokasikan kebutuhan sistem perangkat keras dan perangkat lunak untuk membentuk keseluruhan arsitektur sistem. Tahapan ini mencakup peningkatan, pembaharuan, penambahan dalam pengumpulan data.

3. *Implementation*

   Tahap ini merupakan tahap merubah desain yang telah dibuat sebelumnya di tahap *acquisition*. Tahapan ini mengimplementasikan rangkaian program atau unit program, proses blockchain dan pengujian sistem dengan *tools wireshark* dan *networkminer*. Tahapan ini mencakup proses verifikasi kinerja perangkat *hardware* dan *software*.

4. Operation atau maintance

   Tahapan ini sistem telah diinstal dan benar-benar digunakan. Tahapan ini memastikan performa dari sistem yang sudah dibangun sesuai dengan hasil kalibrasi sebelumnya. Tahapan ini menjelaskan hasil dari pengujian yang dilakukan. Berbeda dengan tahapan sebelumnya, tahapan ini merupakan proses yang berkelanjutan.

5. *Disposition*

   Tahap ini merupakan tahapan untuk memastikan proses pengujian yang diperoleh disimpan secara baik. Sehingga dapat digunakan kembali ketika dibutuhkan.

HASIL PELAKSANAAN PENELITIAN

1. Analisis Kebutuhan

   Analisis alat yang dibutuhkan dalam penelitian ini dapat dilihat pada Tabel 3.1 yang merupakan alat dan bahan yang dibutuhkan dalam penelitian ini yang digunakan untuk memperlancar proses penelitian. Laptop lenovo sebagai tester untuk melakukan pemindaian dalam penelitian ini. Truffle sebagai kerangka pengembangan untuk daaps (aplikasi terdesentralisasi) berdasarkan ethereum blockchain. Ganache yaitu bagian dari trufflesuite yang digunakan untuk menjalankan server ethereum lokal. Solidity digunakan sebagai bahasa pemrograman berorientasi kontrak untuk menulis smart contract. Metamask digunakan sebagai jembatan yang memungkinkan pengguna untuk ke web browser. Rects sebagai library javascript untuk membangun antarmuka pengguna. File hash di blockchain pribadi yang dibangun pada protokol ethereum yang biasa disingkat dengan eth menyimpan file dalam sistem file terdistribusi IPFS. IPFS pada dasarnya yaitu sistem file yang memungkinkan pengguna menyimpan file dan melacak versi dari waktu ke waktu. Sistem penyimpanan ini memungkinkan interaksi langsung melalui jaringan peer-to-peer yang aman. IPFS menggunakan tabel hash terdistribusi digunakan untuk menyimpan data.

2. Pengembangan Sistem Autentikasi Login

   a. **Initiation**

   Simulasi kasus dari penelitian ini menggunakan tool *Visual Studio Code.* Hasil dari tahapan simulasi merupakan sistem yang diuji dan di jalankan. Sistem login memanfaatkan platform *blockchain Ethereum* yang mengimplementasikan teknologi *blockchain* dan *smart contract.* Sebagai antarmuka pemrograman aplikasi, Web3js menghubungkan browser dengan ekstensi yang disebut *metamask,* yang bertindak sebagai jembatan antara sistem login dan *blockchain Ethereum. Metamask* ini bertindak sebagai dompet *Ethereum* untuk manajemen informasi. Sementara itu untuk *smart contract* dibangun dengan menggunakan bahasa pemrograman *solidity*.

   Tahapan penelitian ini user harus menjadi anggota jaringan *blockchain,* ketika jadi anggota maka user akan mempunyai *account ethereum.* Sebuah aplikasi berbasis *blockchain* ethereum harus menjalankan *smart contract*. Sehingga untuk *smart contract* akan di tanda tangani terlebih dahulu oleh orang-orang yang sudah mempunyai *account ethereum* untuk menjalankan sebuah aplikasi berbasis Ethereum. User tersebut selanjutnya jika ingin *login* maka harus sudah terdaftar sebagai penandatangan *smart contract*. Data login dari pengguna akan disimpan sebagai *hash* ke *blockchain* melalui *smart contract*. Pengguna meminta akses ke situs web, misalnya *hash* yang berasal dari kredensial yang diberikan pengguna dibandingkan dengan *hash* yang disimpan di *Smart contract*. Sehingga, saat pengguna melakukan login dan data pengguna cocok maka pengguna berwenang untuk mengakses web. Sebaliknya, jika tidak cocok maka akses menolaknya. Setiap pengguna diharuskan untuk terhubung dengan *address Ethereum* yang

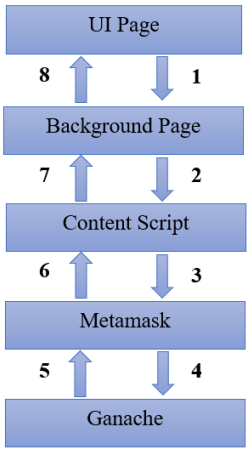
sebelumnya dilakukan dalam proses registrasi, karena *address* ini menghasilkan *hash* login pengguna. Akan tetapi jika *hash* berbeda maka akses ditolak.

b. **Acquisition**

Tahapan ini dilakukan perancangan sistem diantaranya terdapat alur registrasi, login, *flowchart* sistem dan juga desain dari sistem yang akan dibuat.

1) Registrasi

Registrasi menjadi tahapan awal sebelum masuk ke halaman sistem. Pengguna harus menggunakan ekstensi web untuk berinteraksi dengan *metamask* untuk menyebarkan *contract* ke dalam *blockchain Ethereum*. *Metamask* menyediakan *contract* yang akan disebarkan ke jaringan. Begitupun *metamask* juga akan memastikan transaksi ditandatangani. Setelah itu, *metamask* akan menyebarkannya ke jaringan *ethreum*. Alur registrasi dapat dilihat pada Gambar 2.



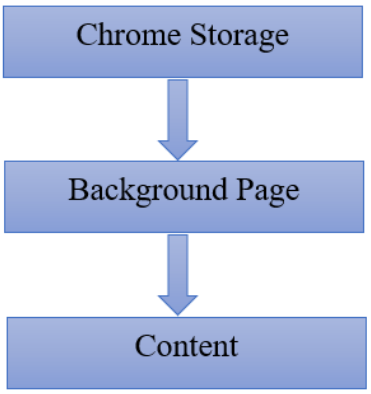
Gambar 2. Alur Registrasi

Gambar 2 menunjukkan alur kerja dari registrasi bagi pengguna yang belum memiliki akun.

- Perintah *deploy* akan dikirimkan ke pengguna
- Selanjutnya *contract* diteruskan ke browser
- Di browser *contract* akan dideteksi oleh *metamask*
- Contract pada *metamask* akan disebarkan setelag melakukan konfirmasi
- Selanjutnya dilakukan pengidentifikasian *contract* dikembalikan saat digunakan
- Pengidentifikasian muncul di situs web
- Setelah pengidentifikasian terdeteksi maka dikembalikan ke ekstensi *metamask*
- Pengidentifikasian kontrak disimpan dalam ekstensi dan sekarang data dapat digunakan.

2) Login

Ketika pengguna sudah memiliki akun, pengguna akan melakukan proses login. Gambar 3 merupakan alur dari login.

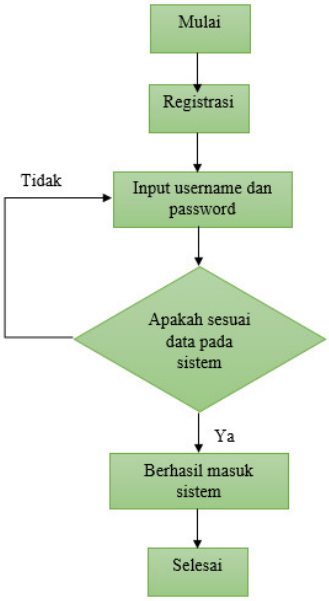


Gambar 3. Alur prosedur Login

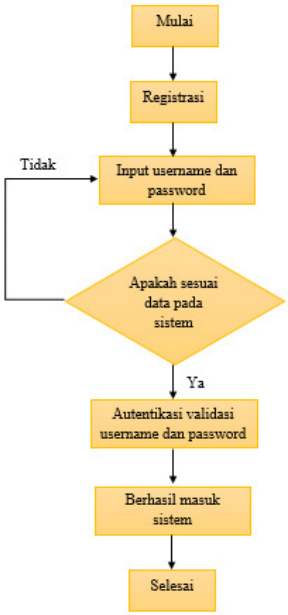
Gambar 3 menunjukkan alur prosedur login, setelah proses registrasi selesai, selanjutnya pengguna akan dialihkan ke halaman login di chrome. Pada chrome storage pengidentifikasian kontrak dari penyimpanan ekstensi *metamask*. Selanjutnya pada *background page* berisi kontrak, stemple waktu dan tanda stemple waktu akan diteruskan ke skrip konten.

3) *Flowchart* Sistem

Flowchart sistem merupakan alur kerja dari penelitian yang dibangun. Gambar 4 dan Gambar 5 merupakan perancangan dari flowchart sistem.



Gambar 4. Flowchart Sistem Sebelum Menggunakan Blockchain



Gambar 5. Flowchart Sistem SetelahMenggunakan Blockchain

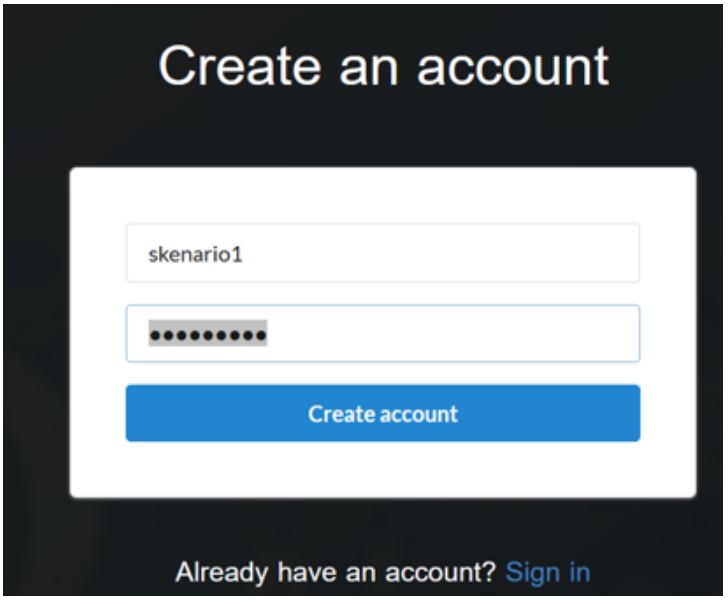
Gambar 4 dan Gambar 5 pada tahap *flowchart* sistem, pengguna akan melakukan login terlebih dahulu. Setelah user login, masukkan *username* dan *password*. Gambar 4 menunjukkan alur sistem sebelum menggunakan teknologi *blockchain* yang tidak melalui proses autentikasi. Sehingga proses tersebut rentan di bobol. Berbeda dengan Gambar 5 dimana sistem akan mengecek apakah data di sistem *blockchain* sudah benar, jika data yang dimasukkan benar, sistem akan melakukan verifikasi *username* dan *password.* Setelah semua proses selesai, pengguna akan masuk ke sistem.

c. **Implementation**

Tahapan ini digunakan untuk mensimulasikan proses *blockchain* dan juga pengujian sistem dengan menggunakan tools *wireshark* dan *networkminer*.

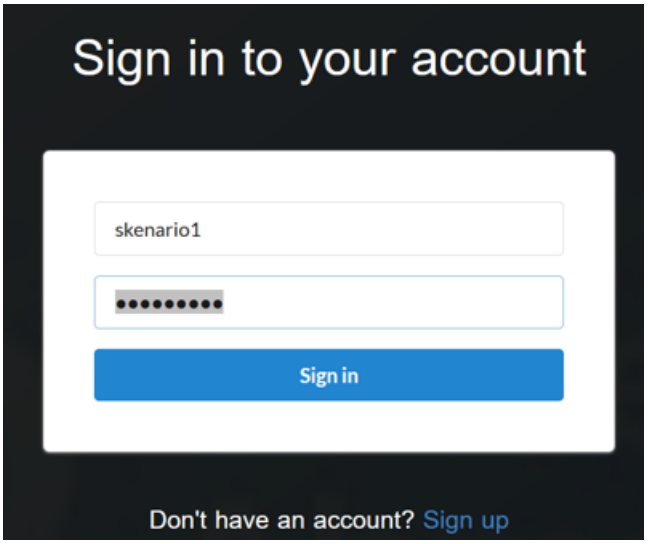
1) Tampilan Aplikasi Sistem Login

Implementasi pada penelitian ini yaitu sistem login dibuat dengan menggunakan Visual Studio Code. Hasil tahapan implementasi merupakan sebuah sistem yang siap diuji dan dijalankan. Halaman registrasi dan login dari sistem login sendiri terdiri dari username, password. Halaman awal dari sistem yang dibangun merupakan langkah pertama yang akan ditampilkan sebelum masuk ke halaman login berhasil atau home. Halaman ini dibuat untuk memberikan batasan kepada pihak yang tidak berkepentingan agar dibuat untuk memberikan batasan kepada pihak yang tidak berkepentingan agar tidak dapat mengakses dan mengolah data tanpa melakukan login terlebih dahulu. Sehingga sebelum login pengguna haru melakukan registrasi untuk mendapatkan akun. Berikut Gambar 6 Tampilan antarmuka berisi form registrasi.



Gambar 6. Tampilan Halaman Registrasi

Gambar 6 menunjukkan data-data yang harus diinputkan. Data tersebut berupa username, password. Setelah melakukan registrasi maka user atau pengguna akan mendapatkan notifikasi berupa pendaftaran berhasil. Kemudian pengguna melakukan login, dapat dilihat pada Gambar 7 Tampilan antarmuka berisi form registrasi login.



Gambar 7. Tampilan Halaman Login

Gambar 7 merupakan tampilan login yang digunakan untuk menginputkan username, password. Apabila yang dimasukkan sudah benar selanjutnya pengguna akan diarahkan ke Dashboard Sistem.

2) Implementasi Blockchain

Sistem *Login* pada penelitian ini menggunakan *platform blockchain Ethereum* yang mengimplementasikan teknologi *blockchain* dan kontrak pintar (*smart contract*). Sebagai antarmuka pemrograman aplikasi, web3js menghubungkan browser dengan ekstensi yang disebut *metamask,* yang bertindak sebagai jembatan antara sistem login dan *blockchain ethereum. Metamask* ini bertindak sebagai dompet *ethereum* untuk manajemen informasi. Sementara itu untuk *smart contract* dibangun dengan menggunakan bahasa pemrograman solidity. Pada kode program 1 tertera skrip program *blockchain* menggunakan solidity

Kode Program 1 Penyimpanan *Hash* data Login

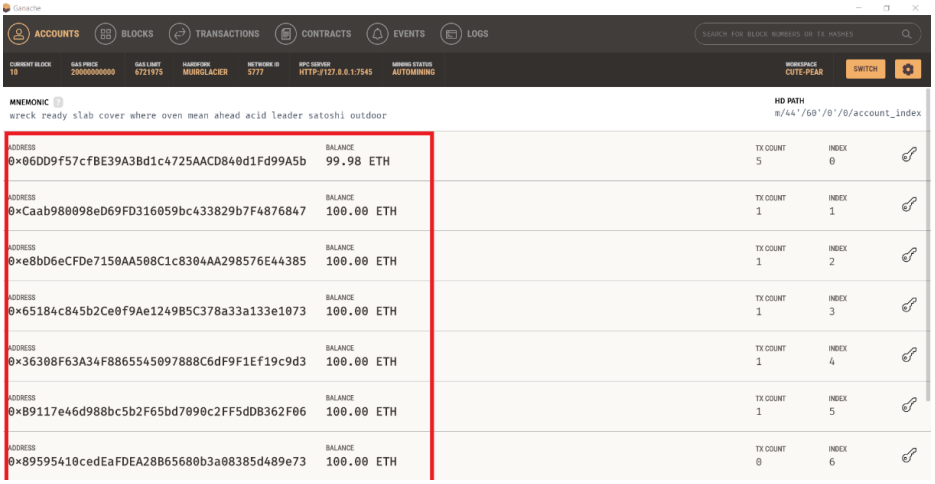
```
1. pragma solidity 0.8.6;
2. contract Authentication {
3. uint256 public nbOfUsers;
4. struct User {
5.  string signature_Hash;
6.   address userAddress;
7.  }
8. //memetakan alamat dompet pengguna dengan ID pengguna
9. mapping(address => User) private user;
10.          constructor() {
11.         nbOfUsers = 0;
12.      }
13.          function register(string memory _signature) public {
14.       require(
15.           user[msg.sender].userAddress ==
16.                address(0x0000000000000000000000000000000000000000),
17.            "already registered"
18.        );
19.              user[msg.sender].signature_Hash = _signature;
20.        user[msg.sender].userAddress = msg.sender;
21.        nbOfUsers++;
22.      }
```

```
23.          function getSignatureHash() public view returns (str
   ing memory) {
24.        require(msg.sender == user[msg.sender].userAddress, "N
   ot allowed");
25.        return user[msg.sender].signatureHash;}
26.       function getUserAddress() public view returns (address)
    {
27.        return user[msg.sender].userAddress; }}
```

Keterangan:

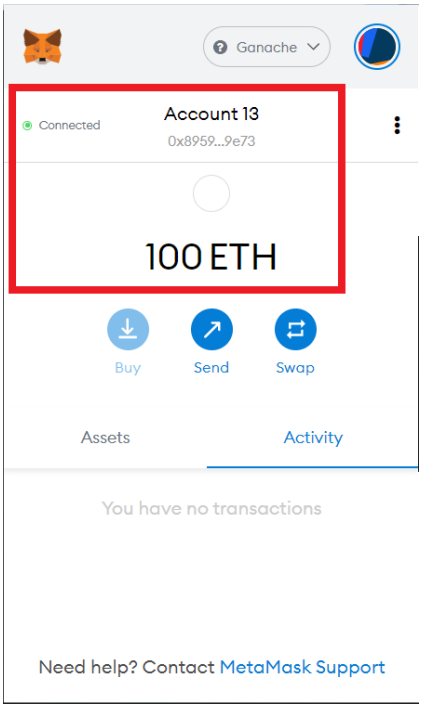
1) Baris 1, solidity versi 0.8.6 pragma yang digunakan untuk mengaktifkan fitur atau pemeriksaan compiler tertentu.
2) Baris 2, untuk mendefinisikan contract
3) Baris 3-7, untuk mendeklarasikan variabel nbofuser dan user
4) Baris 9, untuk memetakan alamat dompet pengguna dengan ID pengguna
5) Baris 10-11, fungsi yang dijalankan sekali dan tidak dapat dipanggil lagi
6) Baris 13, fungsi register dengan parameter signature
7) Baris 14-17, untuk memeriksa user address
8) Baris 19-21, untuk memberikan nilai kedalam variabel signature *hash*, useraddress, dan nbofuser
9) Baris 23, fungsi getsignature*hash* untuk mengembalikan variabel string dari memori
10) Baris 24, untuk memeriksa user address tidak diizinkan
11) Baris 25, untuk mengembalikan nilai variabel user msg. sender. Signature*hash*
12) Baris 26-27, untuk mengembalikan nilai user msg.sender.useraddress

Penelitian ini user atau pengguna harus menjadi anggota jaringan *blockchain*. Terlebih dahulu pengguna membuat akun sebelum mengakses jaringan tersebut. *Ganache* menjadi tempat untuk mendapatkan jaringan. Sebuah aplikasi berbasis *blockchain* ethereum harus menjalankan *smart contract*. *Smart contract* akan di tanda tangani terlebih dahulu oleh orang-orang yang sudah mempunyai ID *Ether* untuk menjalankan sebuah aplikasi berbasis *Ethereum*. Pada *ganache* sudah menyediakan 10 akun default dan setiap akun memiliki saldo 100 eth. Akun tersebut digunakan untuk transaksi di *ganache blockchain*. Berikut Gambar 8 akun default pada *ganache*.



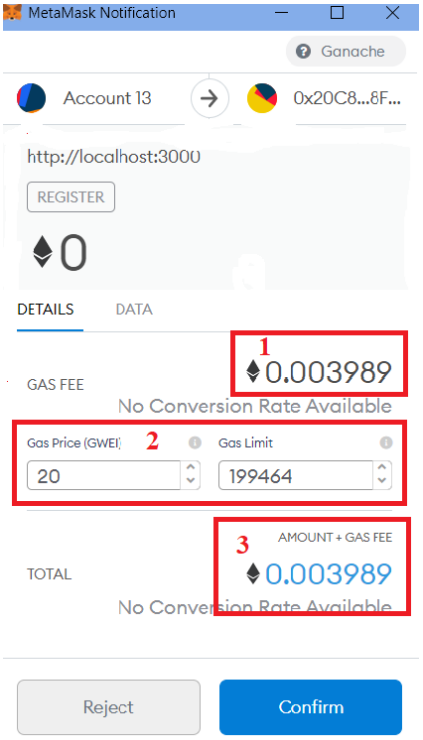
Gambar 8. Akun Address Ethereum

Gambar 8 merupakan tampilan awal dari *ganache*, dimana terdapat 10 *address ethereum* yang memiliki 100 *eth* yang bisa di gunakan untuk simulasi *blockchain*. Satu jaringan ethereum ini digunakan untuk satu proses registrasi. Sehingga apabila pengguna ingin mendaftar akun baru untuk masuk ke dalam sistem login maka pengguna mengambil satu jaringan ethereum yang ada pada *ganache*. Pengguna tersebut jika ingin login maka harus sudah terdaftar sebagai penandatangan *smart contract.* Sebelum pengguna login ke halaman sistem, pengguna terlebih dahulu melakukan *connect ganache* ke *metamask. Metamask* ini sebagai jembatan untuk pengguna masuk ke web browser. *Metamask* ini memungkinkan pengguna untuk menjalankan *ethereum daaps* langsung di browser. Berikut Gambar 4.10 yang menunjukkan akun *metamask* yang sudah terhubung dengan jaringan ethereum. akun ini yang akan digunakan untuk melakukan registrasi.



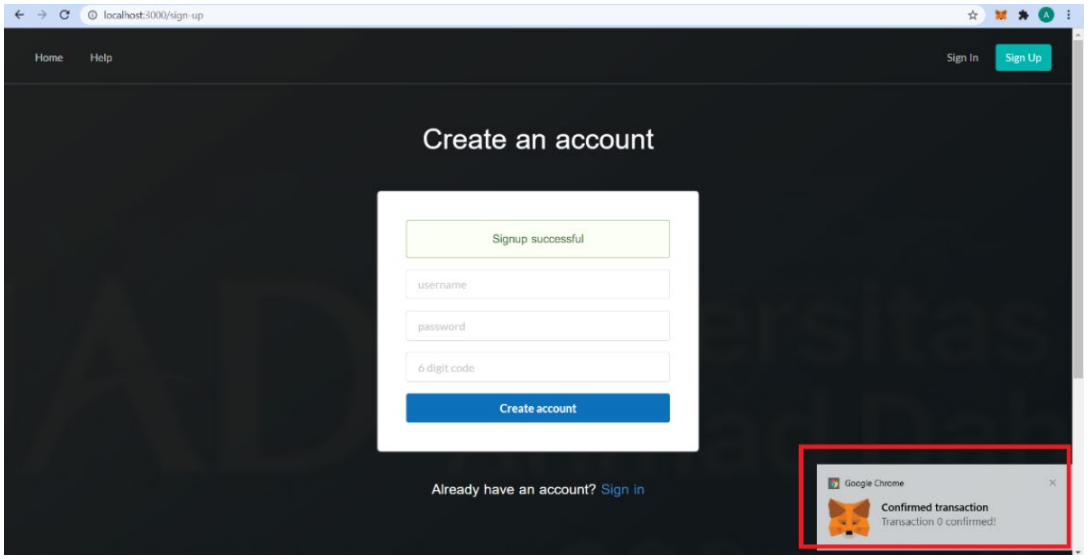
Gambar 9. Akun Metamask yang Sudah Terhubung

Gambar 9 menunjukkan akun *metamask* yang sudah terhubung dengan jaringan *ethereum*. Setelah itu, pengguna kemudian melakukan proses registrasi agar bisa login masuk ke sistem. Proses registrasi ditunjukkan pada Gambar 10.

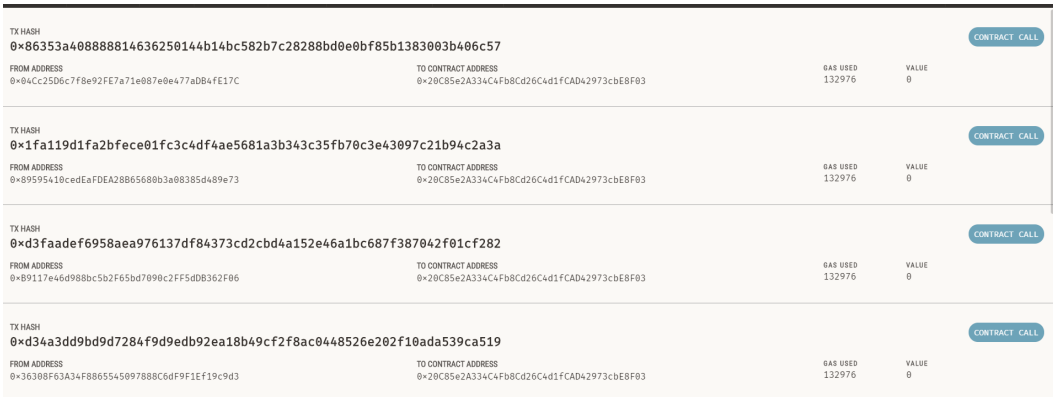


Gambar 10. Proses Registrasi dan Transaksi Blockchain

Gambar 10 diatas pada kotak 1 Setelah pembayaran berhasil, transaksi akan di mining oleh sejumlah node pada jaringan *blockchain*. menunjukkan detail transaksi, dimana *Gas Fee* (kotak 1) merupakan biaya transaksi yang dibayarkan sebesar 0.003989. *Gas Price* (kotak 2) yang dibayarkan sebesar 20 gwei yang artinya transaksi akan diproses dengan cukup cepat. Sedangkan *Gas Limit* (kotak 2) s ebesar 199464. *Amount + Gas Fee* merupakan jumlah yang akan dikeluarkan untuk melakukan transaksi (kotak 3). Ketika pengguna memulai transaksi, permintaan tersebut akan masuk ke *blockchain ethereum* tempat penambang melakukan transaksi. Harga yang harus dibayarkan yaitu *Gas Limit* dan juga *Gas Price.* Setelah transaksi berhasil maka akan di catat pada *blockchain* dan pengguna pun dapat melihat gas yang digunakan. Saat transaksi dinyatakan berhasil maka pengguna dapat melakukan login ke sistem dengan menggunakan username, password yang sudah didaftarkan. Setelah transaksi dinyatakan berhasil dapat dilihat pada Gambar 11.



Gambar 11. Tanda Transaksi Berhasil



Gambar 12. Transaksi Berhasil di Catat di Blockchain

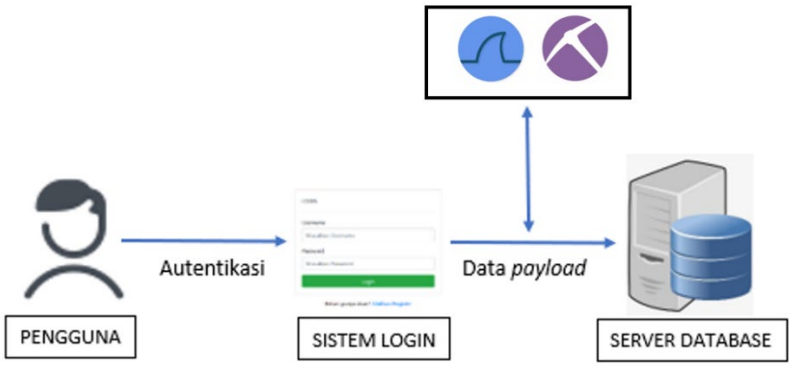
Setelah transaksi berhasil maka akan di catat pada *blockchain* (Gambar 12) dan pengguna pun dapat melihat gas yang digunakan. Saat transaksi dinyatakan berhasil maka pengguna dapat melakukan login ke sistem dengan menggunakan username, password yang sudah didaftarkan. Gambar 13 menunjukkan halaman setelah proses login berhasil.

Gambar 13. Halaman Setelah Login

Gambar 13 menunjukkan akun yang telah didaftarkan oleh pengguna, yang berisikan *address ethereum* dari *blockchain.*

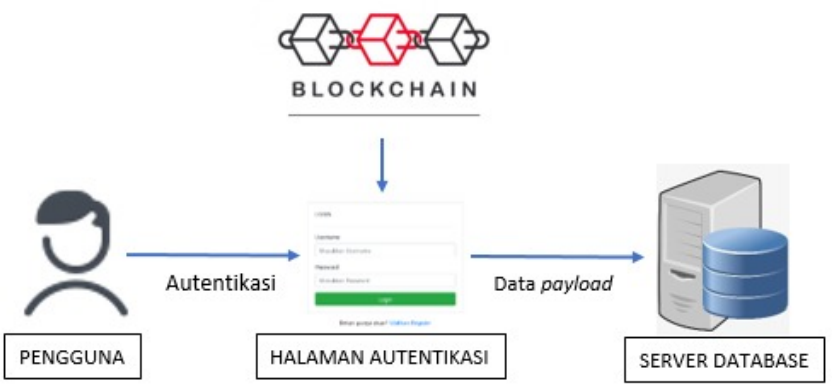
3) Pengujian Sistem

Pengujian sistem dilakukan dengan menggunakan tool *wireshark* dan *networkminer*. Hal ini dilakukan agar mengetahui keamanan dari sistem login yang dibuat dengan menggunakan teknologi *blockchain*. Pengujian dengan menggunakan *wireshark* dilakukan untuk mendapatkan username dan password ketika pengguna sedang melakukan proses login. Pengujian sistem dilakukan dengan dua skenario yaitu sistem login sebelum menggunakan *blockchain* dan setelah menggunakan *blockchain*. Adapun skenario serangan menggunakan menggunakan *wireshark* dapat dilihat pada Gambar 14.



Gambar 14. Skenario Serangan Menggunakan Wireshark dan Networkminer

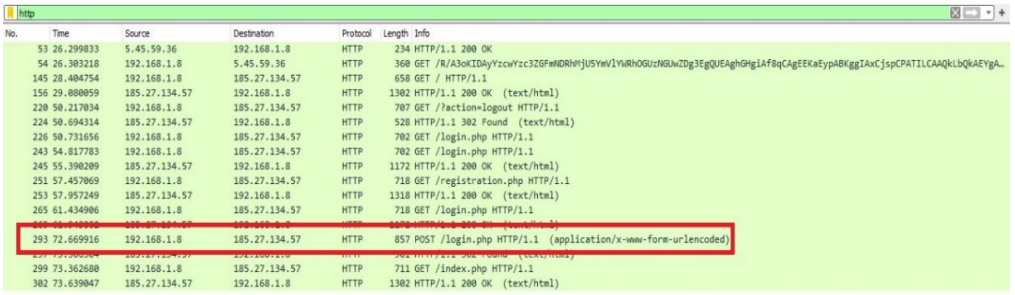
Gambar 14 merupakan skenario yang dilakukan saat pengujian sistem login. Dimana pengguna akan melakukan proses login dengan mengisikan username dan password. Gambar 15 menunjukkan skenario yang dilakukan untuk mengamankan data pengguna dari serangan menggunakan blockchain.



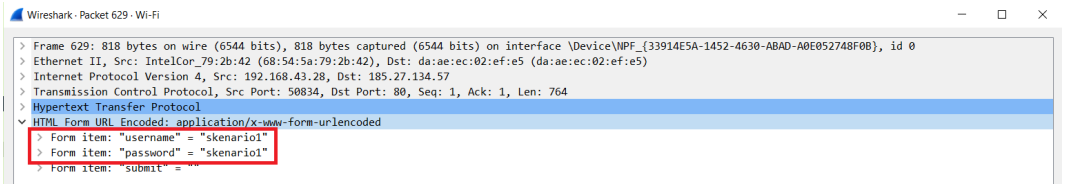
Gambar 15. Skenario Keamanan Sistem Login

Gambar 15 merupakan skenario keamanan yang dilakukan, dimana pengguna melakukan proses pengisian *username* dan *password* kemudian sistem melakukan proses autentikasi. Pengguna jika ingin login maka harus terdaftar sebagai penandatangan *smart contract*. Data login pengguna akan disimpan sebagai hash yang berasal dari kredensial yang diberikan pengguna dibandingkan dengan hash yang

disimpan di *smart contract*. Pengguna melakukan *login* dan data pengguna cocok maka pengguna berwenang untuk mengakses web, sebaliknya jika tidak cocok maka akses menolaknya.
Gambar 14 Saat pengguna melakukan login penyerang pun melakukan capture paket dengan menggunakan *wireshark* untuk bisa mendapatkan file protokol http *Wireshark* akan mengcapture paket-paket yang ditangkap. Untuk mendapatkan file username dan password maka filter paket dengan file protokol HTTP. Sebelum menggunakan teknologi *blockchain*, pengguna akan melakukan login dengan memasukkan username dan password maka penyerang akan dengan mudah mendapatkan data dari pengguna. Hal ini dikarenakan sistem login belum diamankan. Seperti pada Gambar 16 dan Gambar 17.

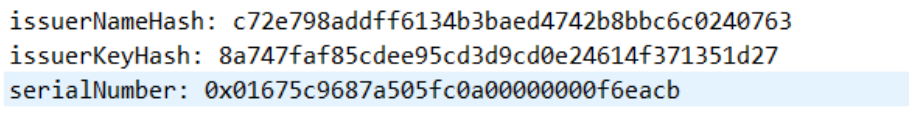


Gambar 16. Filteran Paket HTTP Sebelum Menggunakan *Blockchain*



Gambar 17. File Menampilkan Data Pengguna

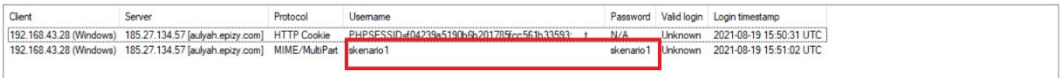
Gambar 17 diatas menggunakan sistem login yang belum di amankan, sehingga keamanan data pengguna dikirimkan melalui jaringan akan mudah dideteksi penyerang, sehingga, untuk mengamankannya pada penilitian ini menggunakan teknologi blockhain yang dapat mengamankan data berupa username dan password dari pengguna. Pengujian sistem login setelah menggunakan *blockchain* dapat dilihat pada Gambar 18.



Gambar 18. Filteran Paket HTTP Setelah Menggunakan *Blockchain*

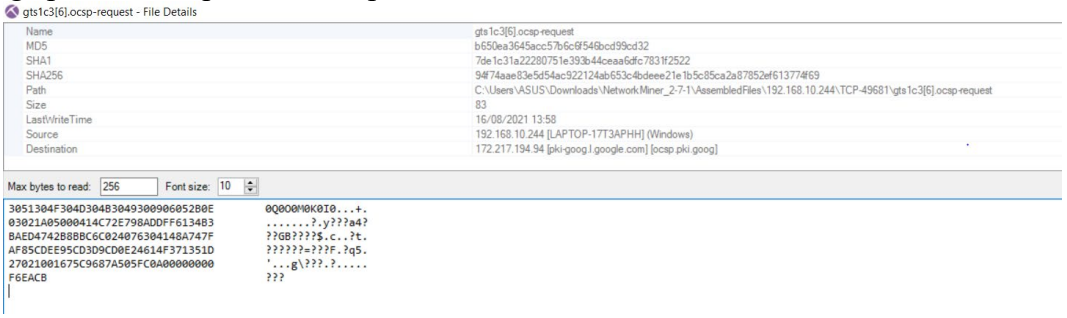
Gambar 18 menunjukkan hasil filteran paket http setelah menggunakan *blockchain*. Apabila data login telah ditambahkan teknologi *blockchain* maka hasil dari analisis paket capture http dari *wireshark* adalah data yang telah diubah ke dalam bentuk *hash* sehingga bagi penyerang tidak dapat melihat username dan password yang terkirim.
Setelah melakukan pengujian menggunakan *wireshark* , selanjutnya akan dilakukan pengujian yang kedua yaitu dengan menggunakan *networkminer*. Data hasil sama

dengan yang sebelumnya menggunakan sistem login sebelum dan sesudah menggunakan teknologi *blockchain*. File dari *wireshark* disimpan dengan menggunakan format (.pcap) setelah itu akan dilakukan pengujian. Sebelum menggunakan *blockchain* data berupa username dan password dapat dibaca oleh penyerang seperti terlihat pada Gambar 17.



Gambar 19 Hasil *Capture* Sebelum Menggunakan *Blockchain*

Gambar 19 menghasilkan data dari pengguna berupa username dan juga password. Berbeda dengan sistem login yang menggunakan *blockchain*, data dari pengguna tidak dapat dibaca oleh penyerang. Hal tersebut dikarenakan data telah diubah kedalam bentuk *hash* yang menyebabkan penyerang sulit untuk mendapatkan username dan juga password seperti terlihat pada Gambar 18.



Gambar 20 Hasil *Capture* Setelah Menggunakan *Blockchain*

Gambar 20 menunjukkan hasil setelah menggunakan *blockchain*. Data pengguna telah di ubah kedalam bentuk *hash*. Penyerang tidak mendapatkan data-data dari pengguna.

d. **Operation/Maintance**

Setelah melakukan beberapa pengujian didapatkan hasil menggunakan *wireshark* dan *networkminer* seperti pada tabel 1. Pengujian sebelum melakukan implementasi *blockchain* dan setelah mengimplementasikan *blockchain*.

Tabe 1 Hasil Pengujian Menggunakan Wireshark dan Networkminer

Wireshark		*Networkminer*	
Sebelum Implementasi	**Setelah Implementasi**	**Sebelum Implementasi**	**Setelah Implementasi**
Skenario1	c72e798addff613 4b3baed4742b8b bc6c02407638a7 47faf85cdee95cd 3d9cd0e24614f3 71351d2701675c 9687a505fc0a00 000000f6eacb	skenario1	3051304f304d304 b3049300906052b 0e03021a0500041 4c72e798addff613 4b3baed4742b8bb c6c024076304148 a747faf85cdee95c d3d9cd0e24614f3 71351d270210016

			75c9687a505fc0a0 0000000f6eacb
skenario2	c72e798addff613 4b3baed4742b8b bc6c02407638a7 47faf85cdee95cd 3d9cd0e24614f3 71351d2700c8df cd1455532c030a 00000000f6ee84	skenario2	30523050304e304 c304a300906052b 0e03021a0500041 4c72e798addff613 4b3baed4742b8bb c6c024076304148 a747faf85cdee95c d3d9cd0e24614f3 71351d27021100c 8dfcd1455532c03 0a00000000f6ee84
skenario3	c72e798addff613 4b3baed4742b8b bc6c02407638a7 47faf85cdee95cd 3d9cd0e24614f3 71351d276abe75 2b5ae2b0800a00 000000f2c374	skenario3	3051304f304d304 b3049300906052b 0e03021a0500041 4c72e798addff613 4b3baed4742b8bb c6c024076304148 a747faf85cdee95c d3d9cd0e24614f3 71351d2702106ab e752b5ae2b0800a 00000000f2c374
skenario4	c72e798addff613 4b3baed4742b8b bc6c02407638a7 47faf85cdee95cd 3d9cd0e24614f3 71351d2700c8df cd1455532c030a 00000000f6ee84	skenario4	30523050304e304 c304a300906052b 0e03021a0500041 4c72e798addff613 4b3baed4742b8bb c6c024076304148 a747faf85cdee95c d3d9cd0e24614f3 71351d27021100c 8dfcd1455532c03 0a00000000f6ee84
skenario5	c72e798addff613 4b3baed4742b8b bc6c02407638a7 47faf85cdee95cd 3d9cd0e24614f3 71351d276abe75 2b5ae2b0800a00 000000f2c374	Skenario5	3051304f304d304 b3049300906052b 0e03021a0500041 4c72e798addff613 4b3baed4742b8bb c6c024076304148 a747faf85cdee95c d3d9cd0e24614f3 71351d2702106ab e752b5ae2b0800a 00000000f2c374

Dari tabel 1 pengujian yang telah dilakukan menunjukkan kondisi sebelum dan sesudah teknologi *blockchain* diimplementasikan. Sebelum implementasi, sistem menampilkan kerentanan terhadap pencurian data berupa username dan password yang dapat dideteksi menggunakan tools *wireshark* dan *networkminer*. Setelah menerapkan teknologi *blockchain*, data yang dikirimkan menjadi aman karena diubah kedalam bentuk *hash*.

Tabel 2. Daftar *log* yang didapatkan sebelum menggunakan *blockchain*

No	Baris *Log get, post* (NoS)	Baris *Log* Keseluruhan (ReDB)	Keterangan
1	122	779	Skenario1
2	204	490	Skenario2
3	228	245	Skenario3
4	130	113	Skenario4
5	340	114	Skenario5

Tabel 2 menunjukkan hasil log sebelum menggunakan *blockchain,* NoS memiliki jumlah rata-rata baris log GET dan POST 204,8 log, sedangkan untuk ReDB memiliki jumlah rata-rata sebanyak 348 log. Jika dimasukkan ke dalam rumus, sebagai berikut:

$$E = 100 - \left(\frac{U_{BC}\ x\ NoS}{\sqrt{ReDB}}\right)$$

$$E = 100 - \left(\frac{5\ x\ 204.8}{\sqrt{348}}\right)$$

$$E = 100 - \frac{1024}{18{,}65}$$

$$E = 100 - 54{,}90$$

$$E = 45{,}1$$

Hasil pengujian yang didapatkan sebelum menggunakan blockchain didapatkan sebesar 45,1%. Tabel 4.3 daftar log setelah menggunakan blockchain.

Tabel 3 Daftar *Log* yang Didapatkan setelah Menggunakan *Blockchain*

No	Baris *Log blockchain* (NoS)	Baris *Log* Keseluruhan (ReDB)	Keterangan
1	15	12020	Skenario1
2	69	8234	Skenario2
3	51	2352	Skenario3
4	32	3196	Skenario4
5	31	4920	Skenario5

Tabel 4.3 menyajikan hasil log yang didapatkan setelah diterapkan blockchain, NoS memiliki jumlah rata-rata baris log blockchain 39,6 log, sedangkan untuk ReDB memiliki jumlah rata-rata log keseluruhan sebanyak 78,38 log. Jika dimasukkan ke dalam rumus, sebagai berikut:

$$E = 100 - \left(\frac{U_{BC}\ x\ NoS}{\sqrt{ReDB}}\right)$$

$$E = 100 - \left(\frac{5\ x\ 39{,}6}{\sqrt{6.144}}\right)$$

$$E = 100 - \frac{198}{78{,}38}$$

$$E = 100 - 2{,}5$$

$$E = 97{,}5$$

Hasil yang didapatkan setelah menggunakan blockchain sebanyak 97,5%. Hal ini. Pengujian yang dilakukan merujuk pada persamaan $E = 100 - \left(\frac{U_{BC}\ x\ NoS}{\sqrt{ReDB}}\right)$ (Toapanta, 2020).

e. **Disposition**

Sistem login telah diamankan dengan menggunakan teknologi blockchain dan smart contract. Data login pengguna disimpan sebagai hash ke blockchain melalui smart contract. Smart contract ditandatangani terlebih dahulu oleh orang-orang yang sudah mempunyai ID Ether untuk menjalankan aplikasi yang berbasis ethereum. Data yang ada pada setiap blok tidak dapat diubah dan dimanipulasi karna tersimpan pada setiap komputer yang berpartisipasi.

Pengujian pada sistem ini menggunakan tools wireshark dan networkminer. Salah satu bentuk implementasi yang relevan diterapkan untuk mendukung dokumentasi adalah menggunakan pendekatan metada eksternal .pcap untuk menyimpan file wireshark, sehingga data aman untuk pengguna. Tahapan disposition menunjukkan bahwa teknologi blockchain dapat diterapkan pada sistem autentikasi login dalam mengamankan data pengguna. Hal ini dibuktikan dengan keseluruhan data dari pengguna berupa username dan password telah terenkripsi atau diubah kedalam bentuk hash.

Pengujian yang dilakukan menunjukkan bahwa keseluruhan hasil pengujian dari sistem autentikasi login sebelum dan setelah menggunakan blockchain dengan menggunakan dua tools yaitu wireshark dan networkminer menunjukkan hasil yang sama. Hasil tersebut diperoleh berdasarkan pengujian sistem autentikasi login dengan menggunakan tools, skenario, dan sampel yang telah ditentukan pada tahap initiation.

Pengujian yang dilakukan menunjukkan bahwa keseluruhan hasil pengujian dari sistem autentikasi login sebelum dan setelah menggunakan blockchain dengan menggunakan dua tools yaitu wireshark dan networkminer menunjukkan hasil yang sama. Hasil tersebut diperoleh berdasarkan pengujian sistem autentikasi login dengan menggunakan tools, skenario, dan sampel yang telah ditentukan pada tahap initiation.

Status luaran berisi **identitas** dan **status ketercapaian setiap luaran wajib** dan **luaran tambahan** (jika ada) yang dijanjikan. Jenis luaran dapat berupa publikasi, perolehan kekayaan intelektual, hasil pengujian atau luaran lainnya yang telah dijanjikan pada proposal. Uraian status luaran harus didukung dengan **bukti kemajuan** ketercapaian luaran sesuai dengan luaran yang dijanjikan. Lengkapi isian jenis luaran yang dijanjikan serta **lampirkan bukti dokumen** ketercapaian luaran wajib dan luaran tambahan.

STATUS LUARAN

Berdasarkan buku panduan penelitian, universitas ahmad dahlan edisi revisi 2021, maka penelitian ini diharapkan memiliki luaran dan target capaian sebagai berikut:

1. Jurnal Nasional terakreditasi Sinta 4 Jurnal Informatika Sunan Kalijaga (JISKa) Status : (***Accepted*****)**
2. Jurnal Nasional terakreditasi Sinta 2 (Kinetik: *Game Technology, Information System, Computer Network, Computing, Electronics, and Control)* Status *: **(Accepted)***
3. Jurnal Internasional (ESJ: *Emerging Science Journal Q1, SJR 0.76)* Status : ***(In Review)***

Peran Mitra berupa **realisasi kerjasama** dan **kontribusi Mitra** baik *in-kind* maupun *in-cash* (untuk Penelitian Terapan dan Pengembangan). **Bukti pendukung** realisasi kerjasama dan realisasi kontribusi mitra **dilaporkan** sesuai dengan kondisi yang sebenarnya. **Lampirkan bukti dokumen** realisasi kerjasama dengan Mitra.

PERAN MITRA

-

Kendala Pelaksanaan Penelitian berisi **kesulitan** atau **hambatan** yang dihadapi selama melakukan penelitian dan mencapai luaran yang dijanjikan, termasuk **penjelasan jika** pelaksanaan penelitian dan luaran penelitian **tidak sesuai** dengan yang direncanakan atau dijanjikan.

KENDALA PELAKSANAAN PENELITIAN

Kendala yang didapatkan dalam menyusun penelitian ini terdapat pada saat proses mendapatkan data pengujian.

Rencana Tindak Lanjut Penelitian berisi uraian rencana tindaklanjut penelitian selanjutnya dengan melihat hasil penelitian yang telah diperoleh. Jika ada target yang belum diselesaikan pada akhir tahun pelaksanaan penelitian, pada bagian ini dapat dituliskan rencana penyelesaian target yang belum tercapai tersebut.

RENCANA TAHAPAN SELANJUTNYA

Penelitian diimplementasikan pada sistem yang real dan memiliki skalabilitas yang besar.

Daftar Pustaka disusun dan ditulis **berdasarkan sistem nomor** sesuai dengan urutan pengutipan. **Hanya pustaka yang disitasi/diacu** pada laporan kemajuan saja yang dicantumkan dalam Daftar Pustaka.

DAFTAR PUSTAKA

1. Abdillah, M. A., Yudhana, A., & Fadil, A. (2020). Sniffing Pada Jaringan WiFi Berbasis Protokol 802.1x Menggunakan Aplikasi Wireshark. *J-SAKTI (Jurnal Sains Komputer Dan Informatika)*, *4*(1), 1. https://doi.org/10.30645/j-sakti.v4i1.181
2. Alam, A., Zia Ur Rashid, S. M., Abdus Salam, M., & Islam, A. (2018). Towards Blockchain-Based E-voting System. *2018 International Conference on Innovations in Science, Engineering and Technology, ICISET 2018*, 351–354. https://doi.org/10.1109/ICISET.2018.8745613
3. Anakath, A. S., Rajakumar, S., & Ambika, S. (2019). Privacy preserving multi factor authentication using trust management. *Cluster Computing*, *22*, 10817–10823. https://doi.org/10.1007/s10586-017-1181-0
4. Aprialim, F., Adnan, & Paundu, A. W. (2021). Penerapan Blockchain dengan Integrasi Smart Contract pada Sistem Crowdfunding. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *5*(1), 148–154. https://doi.org/10.29207/resti.v5i1.2613
5. Arse, M., & Dubey, J. (2020). A Survey of Internet of Things node's transactions Secure through Blockchain Technology. *International Journal of Computer Applications*, *175*(25), 33–37. https://doi.org/10.5120/ijca2020920796
6. Bragagnolo, S., Rocha, H., Denker, M., & Ducasse, S. (2018). SmartInspect: Solidity smart contract inspector. *2018 IEEE 1st International Workshop on Blockchain Oriented Software Engineering, IWBOSE 2018 - Proceedings*, *2018-Janua*, 9–18. https://doi.org/10.1109/IWBOSE.2018.8327566
7. Chaniago, N., Sukarno, P., & Wardana, A. A. (2021). Electronic document authenticity verification of diploma and transcript using smart contract on ethereum blockchain. *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, *7*(2), 149–163. https://doi.org/10.26594/REGISTER.V7I2.1959
8. Cui, Y., Cui, J., & Hu, J. (2020). A Survey on XSS Attack Detection and Prevention in Web Applications. *ACM International Conference Proceeding Series*, 443–449. https://doi.org/10.1145/3383972.3384027
9. Damai, S., Hu, K., Palit, H. N., Handojo, A., Studi, P., Informatika, T., Industri, F. T., Petra, U. K., & Surabaya, J. S. (2019). Implementasi Blockchain : Studi Kasus e-Voting. *Jurnal Infra Petra*, *031*.

10. Deep, G., Mohana, R., Nayyar, A., Sanjeevikumar, P., & Hossain, E. (2019). Authentication protocol for cloud databases using blockchain mechanism. *Sensors (Switzerland)*, *19*(20), 1–13. https://doi.org/10.3390/s19204444
11. Dhammearatchi, D. (2015). *Use of Network Forensic Mechanism*. *7*(4), 21–36.
12. Efanov, D., & Roschin, P. (2018). The all-pervasiveness of the blockchain technology. *Procedia Computer Science*, *123*, 116–121. https://doi.org/10.1016/j.procs.2018.01.019
13. El-Sofany, H. F. (2020). A new cybersecurity approach for protecting cloud services against DDoS attacks. *International Journal of Intelligent Engineering and Systems*, *13*(2), 205–215. https://doi.org/10.22266/ijies2020.0430.20
14. Endicott-Popovsky, B. E., & Frincke, D. A. (2006). Embedding forensic capabilities into networks: Addressing inefficiencies in digital forensics investigations. *Proceedings of the 2006 IEEE Workshop on Information Assurance*, *2006*, 133–139. https://doi.org/10.1109/iaw.2006.1652087
15. Endicott-Popovsky, B., Frincke, D. A., & Taylor, C. A. (2007). A theoretical framework for organizational network forensic readiness. *Journal of Computers*, *2*(3), 1–11. https://doi.org/10.4304/jcp.2.3.1-11
16. Fadlil, A., Riadi, I., & Nugrahantoro, A. (2020). Data Security for School Service Top-Up Transactions Based on AES Combination Blockchain Technology. *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, *11*(3), 155. https://doi.org/10.24843/lkjiti.2020.v11.i03.p04
17. Fat, J., Candra, H., & Wiliam, W. (2019). Sekuritisasi Data Sensor Pada Aplikasi Internet of Things (IoT) Dengan Menggunakan Blockchain Ethereum Di Jaringan Testnet. *TESLA: Jurnal Teknik Elektro*, *21*(1), 79. https://doi.org/10.24912/tesla.v21i1.5886
18. Fauzan N I, A. (2018). *Teknologi Blockchain dan Peranannya Dalam Era Digital*. *4*, 1–15.
19. Gitanjali Simran T, S. D. (2019). *Vulnerability Assessment of Web Applications using Penetration Testing*. *4*, 1552–1556. https://doi.org/10.35940/ijrte.B2133.118419
20. Gupta, Shashank, & Gupta, B. B. (2017). Cross-Site Scripting (XSS) attacks and defense mechanisms: classification and state-of-the-art. *International Journal of Systems Assurance Engineering and Management*, *8*, 512–530. https://doi.org/10.1007/s13198-015-0376-0
21. Gupta, Suyash, & Sadoghi, M. (2020a). Encyclopedia of Big Data Technologies. *Encyclopedia of Big Data Technologies*, *May*. https://doi.org/10.1007/978-3-319-63962-8
22. Gupta, Suyash, & Sadoghi, M. (2020b). Encyclopedia of Big Data Technologies. *Blockchain Transaction Processing*, *May*. https://doi.org/10.1007/978-3-319-63962-8
23. Harahap, E. P., Aini, Q., & Anam, R. K. (2019). Pemanfaatan Teknologi Blockchain Pada Platform Crowdfunding. *Technomedia Journal*, *4*(2), 199–210. https://doi.org/10.33050/tmj.v4i2.1108
24. Hidayat, T. N., & Riadi, I. (2021). Optimation Wireless Security IEEE 802.1X using the Extensible Authentication Protocol-Protected Extensible Authentication Protocol (EAP-PEAP). *International Journal of Computer Applications*, *174*(11), 25–30. https://doi.org/10.5120/ijca2021920988

25. Ismanto, L., Ar, H. S., Fajar, A. N., Sfenrianto, & Bachtiar, S. (2019). Blockchain as E-Commerce Platform in Indonesia. *Journal of Physics: Conference Series*, *1179*(1). https://doi.org/10.1088/1742-6596/1179/1/012114
26. Khera, Y., Kumar, D., Sujay, S., & Garg, N. (2019). Analysis and Impact of Vulnerability Assessment and Penetration Testing. *Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon 2019*, 525–530. https://doi.org/10.1109/COMITCon.2019.8862224
27. Kiran, K. V. V. N. L. S., Devisetty, R. N. K., Kalyan, N. P., & Mukundini, K. (2020). ScienceDirect ScienceDirect ScienceDirect Building a Intrusion Detection System for IoT Environment using Machine Learning Techniques Building a Intrusion Detection System for IoT Environment using Machine Learning Techniques. *Procedia Computer Science*, *171*(2019), 2372–2379. https://doi.org/10.1016/j.procs.2020.04.257
28. Kunang, Y. N., & Ibadi, T. (2013). Celah Keamanan Sistem Autentikasi Wireless Berbasis RADIUS. *Celah Keamanan Sistem Autentikasi Wireless Berbasis RADIUS*, *34*(2), 1907–5022. https://doi.org/DOI:10.13140/RG.2.1.2115.0323
29. Linoy, S., Mahdikhani, H., Ray, S., Lu, R., Stakhanova, N., & Ghorbani, A. (2019). Scalable privacy-preserving query processing over ethereum blockchain. *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, 398–404. https://doi.org/10.1109/Blockchain.2019.00061
30. Marques, N., Zúquete, A., & Barraca, J. P. (2019). *Integration of the Captive Portal paradigm with the 802.1X architecture*. 1–28. http://arxiv.org/abs/1908.09927
31. Mohanta, B. K., & Jena, D. (2018). An Overview of Smart Contract and Use Cases in Blockchain Technology: 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT). *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 1–4.
32. Mokbal, F. M. M., Dan, W., Imran, A., Jiuchuan, L., Akhtar, F., & Xiaoxi, W. (2019). MLPXSS: An Integrated XSS-Based Attack Detection Scheme in Web Applications Using Multilayer Perceptron Technique. *IEEE Access*, *7*, 100567–100580. https://doi.org/10.1109/ACCESS.2019.2927417
33. Noorsanti, R. C., Yulianton, H., & Hadiono, K. (2018). Blockchain - Teknologi Mata Uang Kripto (Crypto Currency). *Prosiding SENDI_U*, *3*(November), 306.
34. Nurfaizi, M. C., Bhawiyuga, A., & Amron, K. (2019). *Pengembangan Gateway untuk Menghubungkan Jaringan IoT (Internet Of Things) Dan Jaringan Blockchain*. *3*(12), 10949–10958.
35. Pallavi, C., Girija, R., & Jayalakshmi, S. L. (2021). An Analysis on Network Security Tools and Systems. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3833455
36. Patel, K. (2019). A survey on vulnerability assessment penetration testing for secure communication. *Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019*, *Icoei*, 320–325. https://doi.org/10.1109/ICOEI.2019.8862767

37. Putra, A. W. P., Bhawiyuga, A., & Data, M. (2018). Implementasi Autentikasi JSON Web Token (JWT) Sebagai Mekanisme Autentikasi Protokol MQTT Pada Perangkat NodeMCU. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *2*(2), 584–593. http://j-ptiik.ub.ac.id
38. Rahardja, U., Aini, Q., Yusup, M., & Edliyanti, A. (2020). Penerapan Teknologi Blockchain Sebagai Media Pengamanan Proses Transaksi E-Commerce. *Computer Engineering, Science and System Journal*, *5*(1), 28–32. https://doi.org/10.24114/CESS.V5I1.14893
39. Rahardja, U., Harahap, E. P., & Christianto, D. D. (2019). Pengaruh Teknologi Blockchain Terhadap Tingkat Keaslian Ijazah. *Technomedia Journal*, *4*(2), 211–222. https://doi.org/10.33050/tmj.v4i2.1107
40. Riadi, I., Umar, R., Aziz, M. A., Informatika, S. T., & Dahlan, U. A. (2021). *Komparatif Web-based Instant Messaging Vulnerability Menggunakan*. *1*(10), 813–819.
41. Riadi, I., Umar, R., & Busthomi, I. (2020). *Optimasi Keamanan Autentikasi dari Man in the Middle Attack (MiTM) Menggunakan Teknologi Blockchain*. *04*, 15–19.
42. Riadi, I., Umar, R., & Lestari, T. (2020). Analisis Kerentanan Serangan Cross Site Scripting (XSS) pada Aplikasi Smart Payment Menggunakan Framework OWASP. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *5*(3), 146. https://doi.org/10.14421/jiska.2020.53-02
43. Riadi, I., Umar, R., & Nasrulloh, I. M. (2018). Analisis Forensik Digital Pada Frozen Solid State Drive Dengan Metode National Institute of Justice (Nij). *Elinvo (Electronics, Informatics, and Vocational Education)*, *3*(1), 70–82. https://doi.org/10.21831/elinvo.v3i1.19308
44. Richard Sharpe, E. W. (2014). *Wireshark User ' s Guide*. 191.
45. Rizky, A., Kurniawan, S., Gumelar, R. D., Kurniawan, V., & Prakoso, M. B. (2021). Use Of blockchain technology in implementing information system security on education. *BEST (Journal of Biology Education Sains & Technology)*, *4*(1), 62–70.
46. Rusdan, M., & Sabar, M. (2020). Analisis dan Perancangan Jaringan Wireless Dengan Wireless Distribution System Menggunakan User Authentication Berbasis Multi-Factor Authentication. *JOINT (Journal of Information Technology)*, *02*(01), 17–24.
47. Salman, T., Zolanvari, M., Erbad, A., Jain, R., & Samaka, M. (2019). Security services using blockchains: A state of the art survey. *IEEE Communications Surveys and Tutorials*, *21*(1), 858–880. https://doi.org/10.1109/COMST.2018.2863956
48. Sarmah, U., Bhattacharyya, D. K., & Kalita, J. K. (2018). A survey of detection methods for XSS attacks. *Journal of Network and Computer Applications*, *118*, 113–143. https://doi.org/10.1016/j.jnca.2018.06.004
49. Shajina, A. R., & Varalakshmi, P. (2017). A novel dual authentication protocol (DAP) for multi-owners in cloud computing. *Cluster Computing*, *20*(1), 507–523. https://doi.org/10.1007/s10586-017-0774-y
50. Shorman, S., Allaymoun, M., & Hamid, O. (2019). Developing the E-Commerce Model a Consumer To Consumer Using Blockchain Network Technique. *International Journal of Managing Information Technology*, *11*(02), 55–64. https://doi.org/10.5121/ijmit.2019.11204

51. Sikos, L. F. (2018). AI in Cybersecurity. In *Springer*. https://doi.org/10.1007/978-3-319-98842-9
52. Singh, S., Sanwar Hosen, A. S. M., & Yoon, B. (2021). Blockchain Security Attacks, Challenges, and Solutions for the Future Distributed IoT Network. *IEEE Access*, *9*, 13938–13959. https://doi.org/10.1109/ACCESS.2021.3051602
53. Singla, V., Malav, I. K., Kaur, J., & Kalra, S. (2019). Develop Leave Application using Blockchain Smart Contract. *2019 11th International Conference on Communication Systems and Networks, COMSNETS 2019*, *2061*, 547–549. https://doi.org/10.1109/COMSNETS.2019.8711422
54. Subekti, Z. M., & Subandri, S. (2020). Implementasi Metode Per Connection Queue Dengan Access User Direct Mac Filtering Pada Jaringan Wireless. *INOVTEK Polbeng - Seri Informatika*, *5*(2), 240. https://doi.org/10.35314/isi.v5i2.1472
55. Susianto, D., Rachmawati, A., Informatika, J. M., Minner, N., Cendikia, C., Lampung, B., & Industri, F. T. (2018). *Implementasi dan Analisis Jaringan Menggunakan Eireshark, Cain and Abels, Network Miner (Studi Kasus : AMIK Dian Cipta Cendikia)*. *XVI*, 120–125.
56. Teferi, F., & Nixon, J. S. (2019). A Security Mechanism to Mitigate DDoS Attack on Wireless Local Area Network (WLAN) using MAC with SSID. *International Journal of Computer Sciences and Engineering*, *7*(4), 864–869. https://doi.org/10.26438/ijcse/v7i4.864869
57. Tian, Y., Zheng, N., Chen, X., & Gao, L. (2021). Wasserstein Metric-Based Location Spoofing Attack Detection in WiFi Positioning Systems. *Security and Communication Networks*, *2021*. https://doi.org/10.1155/2021/8817569
58. Toapanta, S. M., Escalante Quimis, O. A., Mafla Gallegos, L. E., & Maciel Arellano, M. R. (2020). Analysis for the evaluation and security management of a database in a public organization to mitigate cyber attacks. *IEEE Access*, *8*, 169367–169384. https://doi.org/10.1109/ACCESS.2020.3022746
59. Umar, R., Riadi, I., & Zamroni, G. M. (2018). Mobile forensic tools evaluation for digital crime investigation. *International Journal on Advanced Science, Engineering and Information Technology*, *8*(3), 949–955. https://doi.org/10.18517/ijaseit.8.3.3591
60. Vimala, S. T., & Dhas, J. P. M. (2018). SDN based DDoS attack detection system by exploiting ensemble classification for cloud computing. *International Journal of Intelligent Engineering and Systems*, *11*(6), 282–291. https://doi.org/10.22266/IJIES2018.1231.28
61. Wan, L., Eyers, D., & Zhang, H. (2019). Evaluating the impact of network latency on the safety of blockchain transactions. *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, 194–201. https://doi.org/10.1109/Blockchain.2019.00033
62. Wang, S., Yuan, Y., Wang, X., Li, J., Qin, R., & Wang, F. Y. (2018). An Overview of Smart Contract: Architecture, Applications, and Future Trends. *IEEE Intelligent Vehicles Symposium, Proceedings*, *2018-June*(Iv), 108–113. https://doi.org/10.1109/IVS.2018.8500488
63. Wang, Z., Yang, L., Wang, Q., Liu, D., Xu, Z., & Liu, S. (2019). ArtChain: Blockchain-enabled platform for art marketplace. *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, 447–454. https://doi.org/10.1109/Blockchain.2019.00068

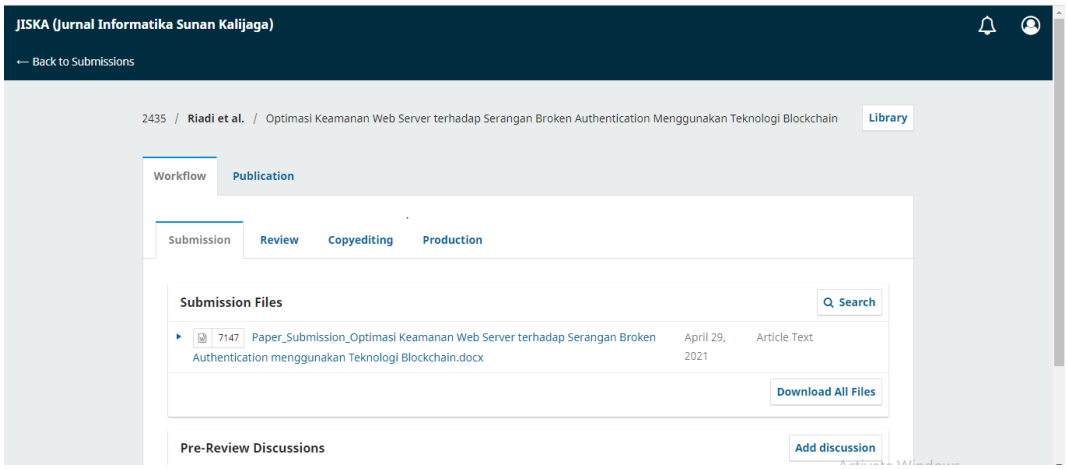
64. Yang, X., Chen, Y., & Chen, X. (2019). Effective scheme against 51% attack on proof-of-work blockchain with history weighted information. *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, 261–265. https://doi.org/10.1109/Blockchain.2019.00041
65. Zhang, R., Xue, R., & Liu, L. (2019). Security and privacy on blockchain. *ArXiv*, *1*(1).
66. Zheng, Y., Li, Y., Wang, Z., Deng, C., Luo, Y., Li, Y., & Ding, J. (2019). Blockchain-based privacy protection unified identity authentication. *Proceedings - 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2019*, 42–49. https://doi.org/10.1109/CyberC.2019.00017

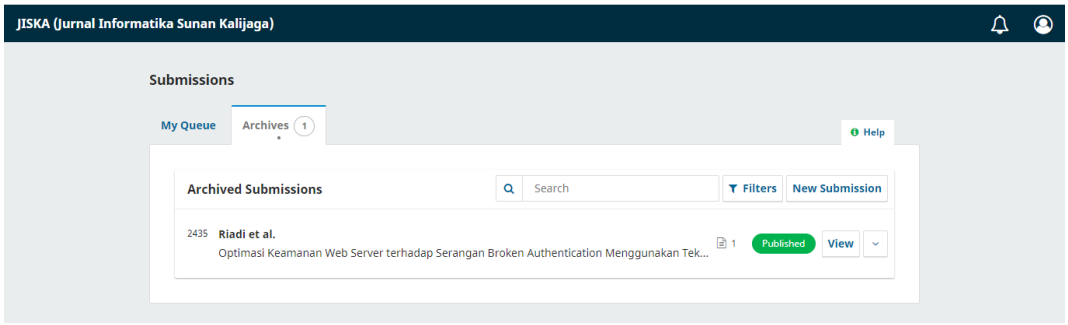
Lampiran-Lampiran

1. Bukti luaran wajib
2. Bukti luaran tambahan (Jika ada)
3. Bukti dokumen realisasi kerjasama dengan mitra (Jika ada)

Lampiran - Lampiran

1. Bukti luaran wajib
   a. Jurnal Jiska





Optimasi Keamanan Web Server terhadap *Serangan Broken Authentication* Menggunakan Teknologi *Blockchain*

Imam Riadi [(1)], Herman [(2)], Aulyah Zakilah Ifani [(3)*]
[1] Sistem Informasi, Fakultas Sains dan Teknologi Terapan, Universitas Ahmad Dahlan, Yogyakarta
[2,3] Teknik Informatika, Fakultas Sains dan Teknologi Terapan, Universitas Ahmad Dahlan,
Yogyakarta
e-mail : imam.riadi@is.uad.ac.id, hermankaha@mti.uad.ac.id,
aulyah1908048022@webmail.uad.ac.id.
* Penulis korespondensi.
Artikel ini diajukan 29 April 2021, direvisi 25 Juni 2021, diterima 11 Juli 2021, dan dipublikasikan 22 September 2021.

Abstract

The aspect of the internet that needs to be considered a security is the login system. The login system usually uses a username and password as an authentication method because it is easy to implement. However, data in the form of usernames and passwords are very vulnerable to theft, so it is necessary to increase the security of the login system. The purpose of this research is to investigate the security of the system. Whether the system is good at protecting user data or not, minimizing execution errors from the system and minimizing risk errors on the system so that the login system can be used safely. This research is conducted to test the system security with Burp Suite *on the login system that has been built. Testing the security of this system by experimenting with POST data which is secured using blockchain technology makes the data sent in the form of hash blocks safer and more confidential so that the system is safer than before. Blockchain technology has successfully secured usernames and passwords from broken authentication attacks. By using the* Burp Suite *testing system, login is more specific in conducting security testing.*

Keywords: Authentication, Broken Authentication, Blockchain, Burp Suite, Login System

Abstrak

Salah satu aspek di internet yang perlu diperhatikan keamanannya adalah sistem *login*. Sistem *login* biasanya menggunakan *username* dan *password* sebagai metode autentikasi karena mudah dalam mengimplementasikannya. Data *username* dan *password* sangat rentan diretas sehingga perlu dilakukan peningkatan keamanan pada sistem *login*. Penelitian bertujuan untuk mengetahui keamanan dari sistem dalam melindungi data pengguna, meminimalkan kesalahan eksekusi dari sistem serta mengurangi risiko *error* pada sistem, sehingga sistem *login* bisa digunakan secara aman. Penelitian ini dilakukan untuk menguji keamanan sistem dengan Burp Suite pada sistem *login* yang dibangun. Pengujian keamanan sistem ini dengan percobaan data POST yang diamankan menggunakan teknologi *blockchain* membuat data yang dikirimkan dalam bentuk blok *hash* menjadi lebih aman dan rahasia sehingga sistem lebih aman daripada sebelumnya. Teknologi *blockchain* berhasil mengamankan *username* dan *password* dari serangan *broken authentication*. Pengujian menggunakan Burp Suite pada sistem *login* lebih spesifik dalam melakukan pengujian keamanan.

Kata Kunci: Autentikasi, Broken Autenthication, Burp Suite, Sistem *Login*, Teknologi *Blockchain*

1. PENDAHULUAN

Keamanan terhadap data, informasi, dan sistem secara keseluruhan semakin penting seiring berkembangnya teknologi informasi. Adanya sistem *login* merupakan salah satu aspek di internet yang perlu diperhatikan keamanannya (Ramadhan & Ariyani, 2018). Kebutuhan informasi dalam internet yang luas memberikan kemudahan dalam mengaksesnya. Data atau informasi menjadi sangat rentan terhadap pencurian sehingga perlu menjaga integritas data ataupun informasi. Dalam mengimplementasikan suatu web diperlukan proses *login*. Sistem keamanan dan proses

sebagaimana tercantum pada

login biasanya menggunakan *username* dan *password* sebagai metode autentikasi. Hal ini digunakan karena kemudahan dalam mengimplementasikannya (Sudiarto Raharjo et al., 2017). Akan tetapi autentikasi menggunakan *username* dan *password* rentan terhadap peretasan. Terutama ketika *username* dan *password* disimpan dalam sebuah *database*. Hal ini terbukti ketika dilakukan *vulnerability assesment* menggunakan beberapa *tools* seperti *openvas,* Burp Suite dan *wireshark*. Kelemahan lainnya, ketika sistem autentikasi diretas, sulit bagi pengguna untuk mengetahuinya (A. W. P. Putra et al., 2018).

Autentikasi adalah suatu pembuktian identitas terhadap suatu entitas seperti pada mesin, kartu kredit, dan orang (Rusdan & Sabar, 2020). Autentikasi dibagi menjadi tiga kategori di antaranya yaitu: *What the entity knows* contoh berupa kata sandi, kedua *What the entity owns* seperti kartu pintar, kunci privasi atau tiket kerberos, dan ketiga *What the entity* yang mencakup teknik autentikasi berdasarkan fitur *biometric* pengguna (sidik jari, bentuk wajah, bentuk tangan, dll.) (Sudiarto Raharjo et al., 2017). Salah satu tekologi inovasi yang mampu menyelesaikan permasalahan tersebut adalah teknologi *Blockchain.*

Blockchain merupakan sebuah teknologi dalam pertukaran informasi tanpa melibatkan pihak ketiga. Informasi berupa informasi dalam bentuk digital, entri data transaksi, aset (Dilley et al., 2016). *Blockchain* adalah teknologi dengan *database* terdistribusi disimpan dan dibagikan ke pengguna yang berwewenang (Parizi et al., 2018) (Bouscaren, 1989). Paling tidak *blockchain* melibatkan tiga unsur teknologi yang sebetulnya sudah ada sejak lama, yaitu internet, protokol dari perangkat lunak, dan kriptografi (Fadlil et al., 2020). Teknologi *blockchain* membuat peretas akan sulit mengubah dan memodifikasi data yang sama di semua komputer di saat yang sama karena membutuhkan waktu yang sangat lama untuk memecahkan kode enkripsi pada setiap blok data di seluruh jaringan komputer (Riadi, Umar, & Busthomi, 2020).

Di balik teknologi *blockchain* terdapat 6 karakteristik utama yaitu *blockchain* adalah kriptografi yang di dalamnya tercatat proses enkripsi yang tingkat keamanan setiap transaksi tinggi. *Blockchain* adalah akuntansi yang di dalamnya tercatat tentang sebuah transaksi. *Blockchain* adalah rantai, ini dikarenakan *blockchain* terdiri dari kumpulan blok, di mana blok sebelumnya harus sama sehingga dapat menyambung seperti sebuah rantai. *Blockchain* merupakan catatan terdistribusi di mana data transaksi dalam *blockchain* tersimpan pada suatu buku besar terdistribusi (distribusi *ledger*) di seluruh

nodes dan sulit dimanipulasi oleh *adversaries*. *Blockchain* adalah *mining* karna setiap seseorang berhasil melakukan validasi kebenaran transaksi maka akan mendapatkan sejumlah imbalan dalam bentuk *native coin* dari *blockchain* tersebut. *Blockchain* adalah *smart contract* karna selain menyimpan data dan transaksi, *blockchain* juga bisa mengeksekusi kontrak perjanjian yang telah disimpan sebelumnya (Rahardja et al., 2019).

Jenis *record* yang terdapat pada sistem *blockchain* yaitu blok dan transaksi. Setiap transaksi *blockchain* tersimpan dalam satu blok secara bersama. Setiap blok membentuk jaringan yang berisi algoritma kriptografi. Algoritma kriptografi digunakan untuk mengambil data dari blok sebelumnya dan diubah ke *Compact String* yang dapat mendeteksi sabotase (Fauzan, 2018). Setiap blok memiliki nilai *hash* yang didapatkan dari blok sebelumnya (Hu et al., 2019). Cara kerja fungsi *hash* yaitu panjang variabel diubah ke dalam bentuk biner. Setiap biner memiliki panjang yang sama. Fungsi *hash* digunakan dalam sistem keamanan, salah satu contohnya autentikasi pesan, penyimpanan *password*, dan tanda tangan digital (H. F. Putra et al., 2019). Untuk mengetahui seberapa besar tingkat keamanan dari suatu sistem penting dilakukan *penetration testing,* berupa simulasi terhadap serangan nyata yang mungkin menyerang sistem tersebut.

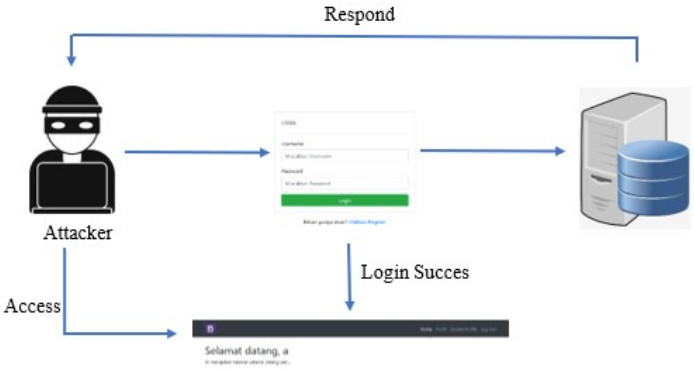
Penetration testing merupakan mengamati serangan dan menganalisis risiko terkait pelanggaran dari keamanan. Penguji tidak hanya dapat mengetahui keberadaan celah bagi *hacker,* tetapi juga dapat mengeksploitasi lebih jauh untuk mengevaluasi tingkat kerentanan sebuah sistem (Azis & Fattah, 2019). *Penetration testing* memerlukan analisa intensif pada kerentanan sistem yang di dapatkan dari kelemahan sistem. Seluruh data analisa yang telah dilakukan akan didokumentasikan dan serahkan ke *user* yang pemilik sistem juga dampak beserta solusi yang didapatkan penguji dari celah keamanan yang ada (Pangalila et al., 2015). Penelitian ini menggunakan Burp Suite untuk menguji keamanan sistem. Burp Suite merupakan *tool* untuk melakukan keamanan *open source* yang digunakan untuk menjalankan dan menguji fitur keamanan pada sebuah aplikasi *website* (Sai Kiran et al., 2020). Burp Suite digunakan untuk menangkap aliran data dengan mengatur sebagai pendengar *proxy* yang bertindak sebagai *server proxy HTTP* lokal (Joshi & Kumar, 2016). Burp Suite secara keseluruhan merupakan kerangka pengujian untuk penetrasi web. Burp Suite terbagi menjadi dua yaitu ada *community edition* dan *profesional edition.* Selain itu Burp Suite sebagai platform terintegrasi berbasis Java untuk melakukan pengujian keamanan suatu aplikasi web. Burp Suite awalnya hanya merupakan aplikasi *proxy server* untuk melakukan *intercept* baik terhadap http-request ataupun httpresponse ke server dan web *application* (T & Sasikala, 2019). Terdapat beberapa kerentanan yang menjadi risiko pada sebuah aplikasi web yaitu injeksi*, insecure direct object references, broken authentication and session management (XSS), security misconfiguration, sensitive data exposure, cross site request forgery (CSRF), cross site scripting (XSS), unvalidated redirects and forwards, using components with known vulnerabilities, missing function level access control* (Guntoro et al., 2020)*.*

Kebanyakan dari pengguna menggunakan pengujian *vulnerability* untuk meningkatkan kesadaran tentang pentingnya keamanan informasi. Kerentanan (*vunlnerability)* dari sebuah sistem disebabkan oleh faktor eksternal dan faktor internal (Wibowo et al., 2019). Dalam menguji *vulnerability* sistem dapat dilakukan dengan 2 tipe yaitu *external testing* dan *internal testing. External Testing* merupakan analisa terhadap informasi *public* yang tersedia. Untuk menampilkan jumlah *network acces point* merupakan *internal testing* yang mewakili beberapa *logical* dan *phisical segment* (Harjowinoto et al.,

2016). metode dalam *vulnerability* yaitu pertama, *passive vulnerability testing* dengan melakukan pengujian terhadap kontrol *login,* konfigurasinya, dan kontrol *web application* sehingga dapat memetakan target sistem. kedua, *active vulnerability testing,* di mana pengujian dilakukan dengan memanipulasi input terhadap kerentanan yang ada, pengambilan hak akses. Ketiga, *aggressive vulnerability testing,* dilakukan *reverse engineering* terhadap *software* dan *system* (Harjowinoto et al., 2016).

Kerentanan (*vulnerability*) banyak diterapkan dan dilakukan contohnya pengujian terhadap web OJS versi 3.0, hasil pengujian didapatkan celah untuk melakukan serangan *Cross-site Scripting* (XSS) (Riadi, Yudhana, et al., 2020). Penelitian juga dilakukan pada sistem administrasi rumah sakit X (Harjowinoto et al., 2016), *security auditing* pada *vulnerable machine* (Sitinjak et al., 2020), dan juga pada pengujian celah keamanan pada CMS (Kunang et al., 2013). Selain menggunakan *OpenVAS* penelitian juga dilakukan pada aplikasi *smart payment* menggunakan *OWASP,* pengujian menggunakan *framework OWASP* menguji kerentanan terhadap serangan XSS (*Cross-site Scripting*) (Riadi, Umar, & Lestari, 2020). Ada beberapa jenis *vulnerability* di antaranya pemindaian berbasis jaringan, pemindaian berbasis *host*, pemindaian jaringan nirkabel, pemindaian aplikasi. Pemindaian berbasis jaringan untuk mengidentifikasi keamanan jaringan. Pemindaian berbasis *host* untuk mengidentifikasi kerentanan server, *host* jaringan lainnya. Pemindaian jaringan nirkabel lebih berfokus pada titik serangan dalam nirkabel. Pemindaian aplikasi untuk mendeteksi kerentanan perangkat lunak. Pemindaian basis data mengidentifikasi titik lemah dalam basis data (Laksmiati, 2020). Dalam penelitian ini menggunakan *broken authentication* sebagai serangan akan digunakan.

Broken authentication merupakan kerentanan web yang terjadi karna kesalahan konfigurasi manajemen *session*. Hal yang harus diperhatikan dari *broken autentication* yaitu pertama, *password strength*, di mana pada aplikasi yang yang kita bangun harus memiliki level minimal dari keamanan *password*, yang dapat dilihat pada panjang *password* dan kompleksitasnya. Kedua, *password use*, di mana aplikasi yang kita buat harus mempunyai batasan *user* mengaksesnya dalam tenggang waktu tertentu. Ketiga, *password storage*, di mana *password* yang kita miliki tidak boleh disimpan dalam aplikasi, dalam hal ini *password* harus ada dalam keadaan terenkripsi. Keempat, *issue* lainnya yang berhubungan contohnya *password* di dalam *source code* tidak boleh dalam bentuk *hard-coded.* Kelima, *Session ID Protection* hal ini digunakan biasanya untuk server mengidentifikasi *user* yang akan masuk ke dalam *session* menggunakan *session ID* (Hassan et al., 2018). *Broken authentication* memiliki fungsi untuk autentikasi dan manajemen *session* yang tidak dapat diterapkan dengan baik, memungkinkan penyerang menyusup untuk mendapatkan *username* dan *password* dan mengeksploitasi kelemahan implementasi untuk mengasumsikan identitas pengguna lain secara permanen atau sementara (OWASP, 2017).



Gambar 1. Skenario Broken *Authentication*.

Gambar 1 menunjukkan *attacker* mengirimkan permintaan kredensial pengguna yang dihasilkan sampai sistem menemukan itu benar. Server memverifikasi kredensial pengguna dan membuat sesi yang kemudian disimpan dalam *database*. Setelah kredensial yang didapat ditebak cocok dengan *database*, sistem mengirimkan respons ke *attacker* atau penyerang dengan akses di akun.

Teknologi *blockchain* digunakan dalam penelitian ini untuk mengamankan data *username* dan *password* pengguna. Sebagaimana diketahui *blockchain* menggunakan fitur tanda tangan digital untuk melakukan transaksi sehingga data dari *user* tidak dapat diubah atau dirusak, hal ini menjadi kelebihan dari *blockchain*. Selain itu, *blockchain* menggunakan sistem terdesentralisasi yang dapat membuat transaksi lebih aman, cepat, dan lancar (Parizi et al., 2018). Sehingga penelitian ini menggunakan *blockchain* sebagai tujuan untuk meningkatkan keamanan *username* dan *password* dari suatu sistem *login*. Pengujian sistem menggunakan *tool* Burp Suite untuk memastikan sistem *login* aman digunakan.

2. METODE PENELITIAN

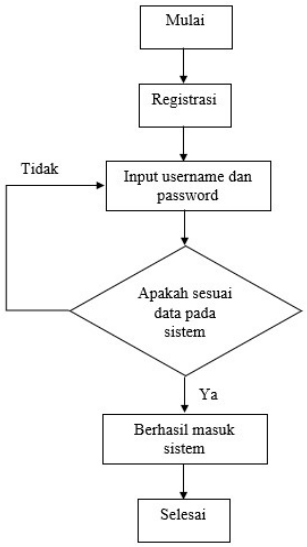
2.1. *Objek Penelitian*

Sistem *login* adalah aplikasi yang digunakan sebagai objek pada penelitian ini. Aplikasi ini merupakan aplikasi atau *website* yang akan digunakan di berbagai aplikasi lainnya. Sistem *login* terdapat *username* dan *password* yang akan diisikan terlebih dahulu sebelum masuk ke sistemnya. Memasukkan *username* dan *password* menjadi sangat rentan terhadap peretasan. Bahnyaknya *tools* yang tersedia membuat data pengguna menjadi terancam. Hal ini tentunya akan sangat berbahaya apabila dibiarkan terus menerus. Oleh karena itu pada penelitian ini akan dilakukan optimasi keamanan. Dengan menggunakan teknologi *blockchain* diharapkan dapat digunakan sebagai pengamanan sistem *login*. Pengujian dalam penelitian ini menggunakan serangan *broken authentication.*

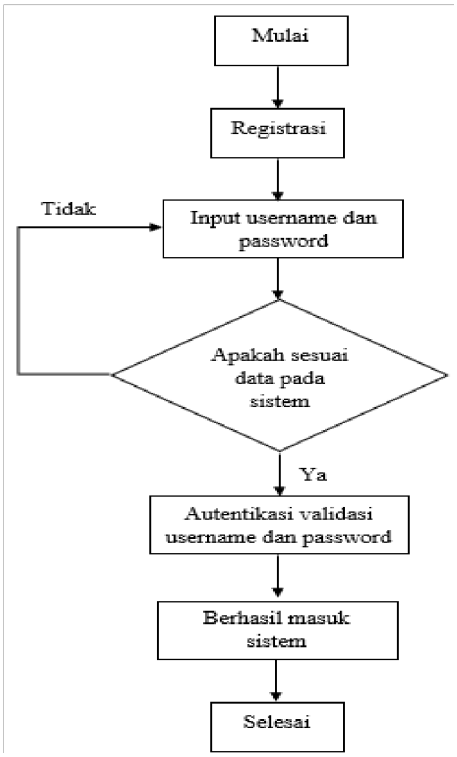
2.2. *Desain Sistem*

Desain *flowchart* sistem menggambarkan alur kerja sistem yang akan dirancang. Sistem mempunyai kemampuan untuk melakukan pengecekan data yang ada pada sistem *login*. Sistem mempunyai kemampuan untuk melakukan autentikasi validasi *username* dan *password*. Sebelum melakukan autentikasi sistem terlebih dahulu melakukan pengecekan data yang ada pada sistem *blockchain*. Tahapan *flowchart* sistem, pertama

pengguna atau *user* akan melakukan *login* terlebih dahulu. Setelah *login* pengguna memasukan *username* dan *password*. Sistem akan mengecek apakah data ada pada sistem *login*, jika data yang dimasukkan sudah benar maka sistem akan melakukan autentikasi validasi *username* dan *password*, setelah semua proses selesai maka pengguna akan masuk sistem. Berikut perancangan *flowchart* sistem dapat dilihat pada Gambar 2.



Gambar 2. *Flowchart* Sistem Sebelum Menggunakan *Blockchain*.



Gambar 3. *Flowchart* Sistem Setelah Menggunakan *Blockchain*.

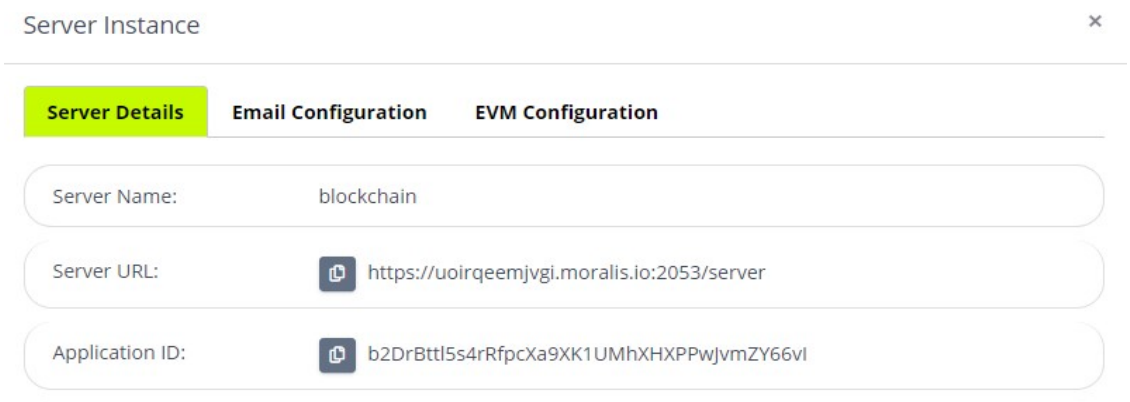
Gambar 2 dan Gambar 3 menunjukkan *flowchart* sistem di mana pengguna atau *user* akan melakukan *login* terlebih dahulu sebelum masuk ke sistem. Setelah memasukan

username dan *password* sistem akan melakukan autentikasi *username* dan *password*. Jika sudah sesuai maka proses *login* akan berhasil. Pada Gambar 2 setelah pengguna melakukan registrasi lalu *login* ke sistem, sistem akan mengecek dan berhasil masuk apabila data yang dimasukkan sesuai. Berbeda dengan *flowchart* yang ada pada gambar 3 yang menggunakan pengecekan autentikasi *username* dan *password* terlebih dahulu sebelum masuk ke dalam sistem, di mana Gambar 3 terdapat proses autentikasi dengan MetaMask. MetaMask digunakan sebagai jembatan antara sistem *login* dengan *blockchain* Ethereum. Sedangkan Gambar 2 tidak melalui proses tersebut sehingga sangat rentan terhadap peretasan.

3. HASIL DAN PEMBAHASAN

Sistem *login* pada penelitian ini memanfaatkan penggunaan *platform blockchain* Ethereum yang mengimplementasikan teknologi *blockchain* dan *smart contract*. Dengan web3.js sebagai *application programming interface* (API) untuk menghubungkan *browser* dengan ekstensi yang dinamakan MetaMask sebagai jembatan antara sistem *login* dengan *blockchain* Ethereum. MetaMask ini bertindak sebagai wallet Ethereum untuk pengelolaan informasi. Sementara itu untuk *smart contract* dibangun dengan menggunakan tools Visual Studio Code (VSCode). Pada sistem *login* juga menggunakan Moralis sebagai autentikasi ke MetaMask. Moralis mengambil informasi dari akun MetaMask. Untuk menghubungkan Moralis dengan MetaMask perlu adanya penyinkronan yang dilakukan di VSCode.

User harus menjadi anggota jaringan *blockchain* ketika jadi anggota maka *user* akan mempunyai *user* ID Ethereum. Sebuah aplikasi berbasis *blockchain* Ethereum harus menjalankan *smart contract*. Sehingga untuk *smart contract* akan ditanda tangani terlebih dahulu oleh orang-orang yang sudah mempunyai ID Ethereum untuk menjalankan sebuah aplikasi berbasis Ethereum. *User* tersebut selanjutnya jika ingin *login* maka harus sudah terdaftar sebagai penandatangan *smart contract*. Berikut merupakan tampilan dari akun Moralis pada Gambar 4.



Gambar 4. Server URL dan *application* ID akun Moralis.

Gambar 4 merupakan server URL dan *application* ID pada akun Moralis yang akan disalin untuk menyinkronkan ke MetaMask. Link URL tersebut kemudian disalin dan dimasukkan ke dalam *source code* di VSCode. Berikut Gambar 5 merupakan URL untuk menyinkronkan akun MetaMask dan Moralis.

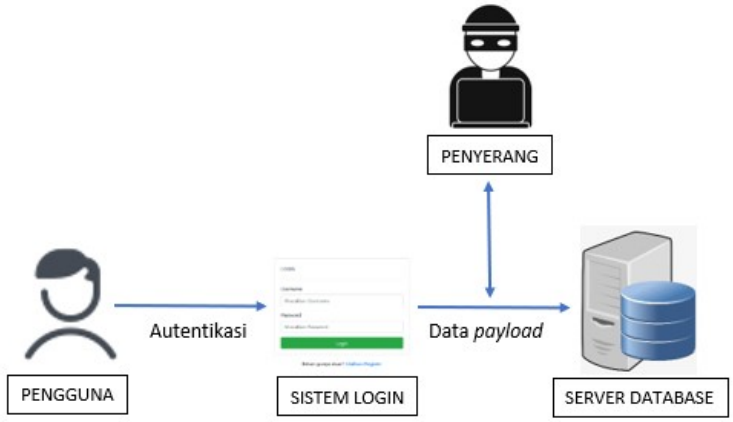
```
<script>
    Moralis.initialize("b2DrBttl5s4rRfpcXa9XK1UMhXHXPPwJvmZY66vI"); // Application id from moralis.io
    Moralis.serverURL = "https://uoirqeemjvgi.moralis.io:2053/server"; //Server url from moralis.io
```

Gambar 5. *Source code* penyinkronan akun MetaMask dan Moralis.

Gambar 5 menunjukkan bagaimana menghubungkan antara akun MetaMask dengan Moralis. Menghubungkan akun tersebut terlebih dahulu masuk ke akun Moralis poc.moralis.io untuk mendapatkan *application* ID dan server URL tersebut. *Application ID from Moralis.io* digunakan untuk menghubungkan web3. Sedangkan server URL digunakan untuk menghubungkan Moralis. Ketika sistem sudah terhubung selanjutnya mengisikan *username* dan *password* dan akan mendapatkan notifikasi dari MetaMask dan diminta untuk memberikan tanda tangan terlebih dahulu.

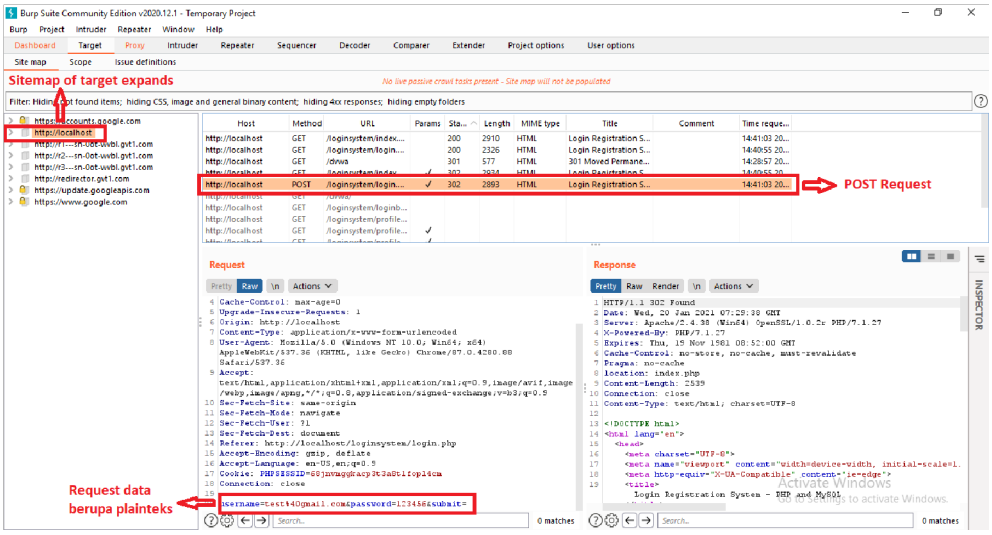
Hasil penerapan teknologi *blockchain* sebagai autentikasi pada sistem *login* mempunyai kemampuan untuk melakukan pengecekan data yang ada pada sistem *blockchain.* Komponenkomponen yang ada pada sistem meliputi menu registrasi, menu *login*, menu lihat profil, menu *update* profil dan *logout.* Halaman registrasi merupakan gambaran awal ketika pengguna belum memiliki akun untuk *login* yang digunakan untuk mendaftarkan *username* dan *password* sebelum masuk ke sistem. Menu *login* merupakan langkah pertama sebelum masuk ke sistem. Halaman ini dibuat untuk memberikan batasan kepada pihak yang tidak berkepentingan agar tidak dapat mengakses dan mengoleh data tanpa melakukan *login* terlebih dahulu. Sehingga sebelum melakukan *login*, pengguna atau *user* harus melakukan registrasi terlebih dahulu untuk mendapatkan akun. *User* memasukkan *username* dan *password* dengan benar maka akan diarahkan ke menu awal dari sistem. Setelah *login* berhasil maka *user* dapat melihat profil dan melakukan *update* data dari *user*.

Pengujian dilakukan menggunakan aplikasi Burp Suite sebagai *software* untuk melakukan percobaan serangan *broken authentication* pada sistem *login*. Autentikasi yang diperlukan yaitu *username* dan *password* yang sudah terdaftar sebelumnya. Autentikasi dengan *username* dan *password* sangat mudah untuk mengimplementasikannya. Akan tetapi, menggunakan *username* dan *password* membuat para peretas mudah untuk melakukan serangan. Gambar 6 akan dipaparkan skenario yang akan dilakukan sebelum menggunakan teknologi *blockchain*.



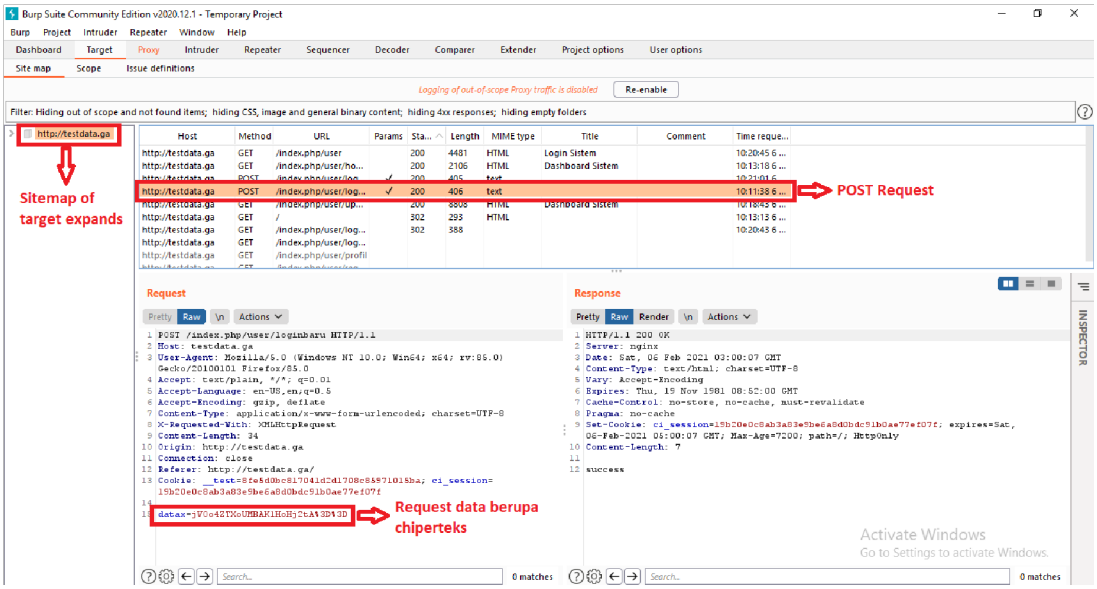
Gambar 6. Skenario keamanan sistem *login*.

Gambar 6 menunjukkan mengenai skenario keamanan yang ada pada sistem. Pengguna akan meginputkan *username* dan *password* ke dalam sistem *login*. Data *payload* akan dikirimkan ke *database* server. Penambahan *blockchain* pada autentikasi sistem *login* membuat penyerang tidak dapat membaca isi *payload* data yang dikirimkan pengguna. Pengujian sistem menggunakan *tool* Burp Suite sebagai *software* untuk melakukan penetrasi pada halaman *login* untuk mencoba masuk ke dalam sistem. Penetrasi ini melakukan bantuan beberapa aplikasi dalam pengujiannya nanti. Pengujian ini difokuskan pada celah keamanan *login* dan autentikasi pada sistem. Percobaan data POST yang diamankan dengan teknologi *blockchain* membuat data yang dikirimkan dalam bentuk blok *hash.* Blok *hash* menjadi lebih aman dan rahasia, data POST dikirimkan ke *database* server. Pengujian sebelum mengunakan *blcokchain* dapat dilihat pada Gambar 7.



Gambar 7. Aplikasi Burp Suite menampilkan *username* dan *password*.

Gambar 7 menunjukkan hasil pengujian setelah diimplementasikannya *blockchain* pada sistem *login*. Aplikasi Burp Suite menampilkan *username* dan *password* ketika masuk ke dalam sistem. Pengujian ini menggunakan *tool* Burp Suite untuk mencari kombinasi *username* dan *password* yang benar untuk masuk ke sistem. Setelah berhasil menangkap status dan proses *login* ini pada sistem bisa dilakukan penetrasi *password*. Hasil pengujian dapat dilihat pada Gambar 8. Pengujian percobaan serangan *broken authentication* pada *tool* Burp Suite menghasilkan data yang dikirimkan berupa blok *hash* atau terenkripsi, sehingga data yang ada dapat terjamin keamanannya dan terjaga rahasianya.



Gambar 8. Capture data setelah menggunakan *Blockchain*.

Gambar 8 menunjukkan hasil *capture* data setelah menggunakan *blockchain*. Data yang didapatkan berupa blok yang terenkripsi. *Username* dan *password* yang dimasukkan sebelumnya berhasil diubah. Serangan dengan menggunakan Burp Suite tidak dapat mendeteksi data dari pengguna setelah menggunakan teknologi *blockchain.*

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, kesimpulan yang dapat diambil yaitu sistem *login* berhasil dibangun. Dengan menggunakan teknologi *blockchain*, data dari *user* terjaga kerahasiaannya. Pengujian dilakukan dengan menggunakan Burp Suite. Data yang dikirimkan berupa blok *hash* dan terenkripsi, sehingga data terjamin keamanannya. Pengujian keamanan sistem *login* ini berhasil dilakukan dengan tepat sehingga sistem lebih aman daripada sebelumnya. Burp Suite sebagai pengujian sistem *login* lebih spesifik dalam melakukan pengujian keamanan. Data *username* dan *password* diubah menjadi chiperteks sehingga penyerang tidak dapat mengetahui data dari pengguna.

DAFTAR PUSTAKA

Azis, H., & Fattah, F. (2019). ANALISIS LAYANAN KEAMANAN SISTEM KARTU TRANSAKSI ELEKTRONIK MENGGUNAKAN METODE PENETRATION TESTING. *ILKOM Jurnal Ilmiah*, *11*(2), 167–174. https://doi.org/10.33096/ilkom.v11i2.447.167-174

Bouscaren, E. (1989). Elementary pairs of models. *Annals of Pure and Applied Logic*, *45*(2), 129– 137. https://doi.org/10.1016/0168-0072(89)90057-2

Dilley, J., Poelstra, A., Wilkins, J., Piekarska, M., Gorlick, B., & Friedenbach, M. (2016). *Strong Federations: An Interoperable Blockchain Solution to Centralized Third-Party Risks*.

Fadlil, A., Riadi, I., & Nugrahantoro, A. (2020). Data Security for School Service Top-Up Transactions Based on AES Combination Blockchain Technology. *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, *11*(3), 155. https://doi.org/10.24843/LKJITI.2020.v11.i03.p04

Fauzan, N. I. (2018). TEKNOLOGI BLOCKCHAINDAN PERANANNYA DALAM ERA DIGITAL.

Jurnal BJB University, *4*, 1–15.

Guntoro, G., Costaner, L., & Musfawati, M. (2020). Analisis Keamanan Web Server Open Journal System (Ojs) Menggunakan Metode Issaf Dan Owasp (Studi Kasus Ojs Universitas Lancang Kuning). *JIPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, *5*(1), 45. https://doi.org/10.29100/jipi.v5i1.1565

Harjowinoto, D., Noertjahyana, A., & Andjarwirawan, J. (2016). Vulnerability Testing Pada Sistem Administrasi Rumah Sakit X. *Jurnal Infra*, *4*(1), 227–232.

Hassan, M. M., Nipa, S. S., Akter, M., Haque, R., Deepa, F. N., Rahman, M. M., Siddiqui, M., & Sharif, M. H. (2018). Broken Authentication and Session Management Vulnerability: A Case Study of Web Application. *International Journal of Simulation: Systems, Science & Technology*, 1–11. https://doi.org/10.5013/IJSSST.a.19.02.06

Hu, S. D. K., Palit, H. N., & Handojo, A. (2019). Implementasi Blockchain: Studi Kasus E-Voting. *Jurnal Infra*, *7*(1), 183–189.

Joshi, C., & Kumar, U. (2016). Security Testing and Assessment of Vulnerability Scanners in Quest of Current Information Security Landscape. *International Journal of Computer Applications*, *145*(2), 1–7. https://doi.org/10.5120/ijca2016910563

Kunang, Y. N., Muklis, F., & Sauda, S. (2013). Pengujian Celah Keamanan Pada Cms (Content Management System). *Prosiding Seminar Nasional Ilmu Komputer (SeNAIK 2013)*, 398–406.

Laksmiati, D. (2020). Vulnerability Assessment Pada Situs Www.Hatsehat.Com Menggunakan Openvas. *Jurnal Akrab Juara*, *5*(3), 240–246.

OWASP. (2017). *OWASP Top Ten Web Application Security Risks*. OWASP. https://owasp.org/www-project-top-ten/

Pangalila, R., Noertjahyana, A., & Andjarwirawan, J. (2015). Penetration Testing Server Sistem Informasi Manajemen Dan Website Universitas Kristen Petra. *Jurnal Infra*, *3*(2), 271–276.

Parizi, R. M., Dehghantanha, A., Choo, K.-K. R., & Singh, A. (2018). Empirical Vulnerability Analysis of Automated Smart Contracts Security Testing on Blockchains. *In Proceedings of the 28th Annual International Conference on Computer Science and Software Engineering (CASCON18)*, 103–113. https://doi.org/10.5555/3291291.3291303

Putra, A. W. P., Bhawiyuga, A., & Data, M. (2018). Implementasi Autentikasi JSON Web Token (JWT) Sebagai Mekanisme Autentikasi Protokol MQTT Pada Perangkat NodeMCU. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK)*, *2*(2), 584–593.

Putra, H. F., Wirawan, W., & Penangsang, O. (2019). Penerapan Blockchain dan Kriptografi untuk Keamanan Data pada Jaringan Smart Grid. *Jurnal Teknik ITS*, *8*(1). https://doi.org/10.12962/j23373539.v8i1.38525

Rahardja, U., Harahap, E. P., & Christianto, D. D. (2019). Pengaruh Teknologi Blockchain Terhadap Tingkat Keaslian Ijazah. *Technomedia Journal*, *4*(2), 211–222. https://doi.org/10.33050/tmj.v4i2.1107

Ramadhan, M. S., & Ariyani, F. (2018). PENINGKATAN KEAMANAN LOGIN WEBSITE DENGAN IMPLEMENTASI ONE TIME PASSWORD MENGGUNAKAN ALGORITMA SHA1 DAN MD5 BERBASIS MOBILE. *SKANIKA*, *1*(2), 689–696.

Riadi, I., Umar, R., & Busthomi, I. (2020). Optimasi Keamanan Autentikasi dari Man in the Middle Attack (MiTM) Menggunakan Teknologi Blockchain. *Journal of Information Engineering and Educational Technology*, *4*(1), 15–19. https://doi.org/http://dx.doi.org/10.26740/jieet.v4n1.p15-19

Riadi, I., Umar, R., & Lestari, T. (2020). Analisis Kerentanan Serangan Cross Site Scripting (XSS) pada Aplikasi Smart Payment Menggunakan Framework OWASP. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *5*(3), 146–152. https://doi.org/10.14421/jiska.2020.53-02

Riadi, I., Yudhana, A., & W, Y. (2020). Analisis Keamanan Website Open Journal System Menggunakan Metode Vulnerability Assessment. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *7*(4), 853. https://doi.org/10.25126/jtiik.2020701928

Rusdan, M., & Sabar, M. (2020). Design and Analysis of Wireless Network with Wireless Distribution System using Multi-Factor Authentication-based User Authentication. *Journal of Information Technology*, *2*(1), 17–24. https://doi.org/10.47292/joint.v2i1.004

Sai Kiran, K. V. V. N. L., Devisetty, R. N. K., Kalyan, N. P., Mukundini, K., & Karthi, R. (2020). Building a Intrusion Detection System for IoT Environment using Machine Learning Techniques. *Procedia Computer Science*, *171*(2019), 2372–2379. https://doi.org/10.1016/j.procs.2020.04.257

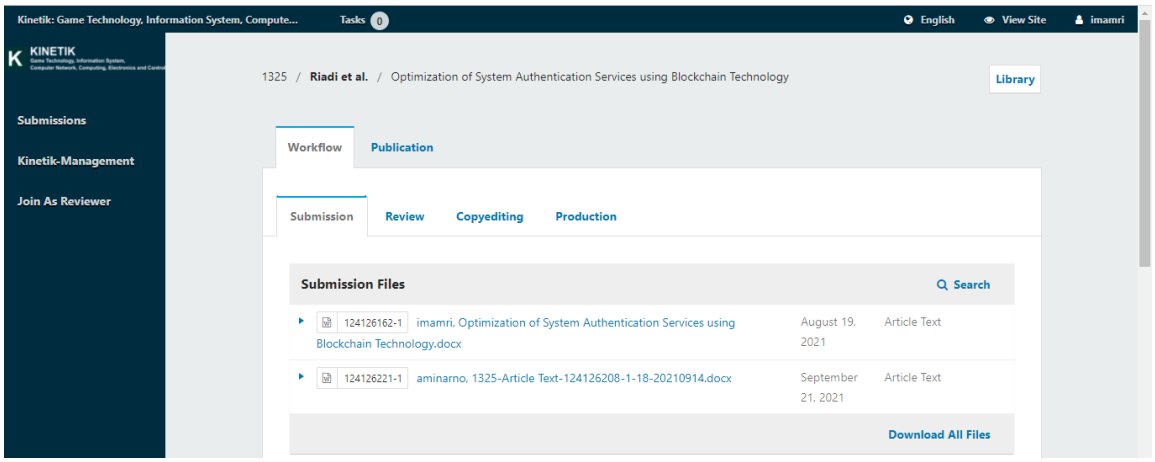
Sitinjak, H. S. F., Hediyanto, U. Y. K. S., & Widjajarto, A. (2020). Security Auditing Pada Vulnerable Machine Menggunakan Open Source Ids Dan Vulnerability Scanner Berdasarkan Nist Cybersecurity Framework. *EProceedings of Engineering*, *7*(2), 7638– 7646.

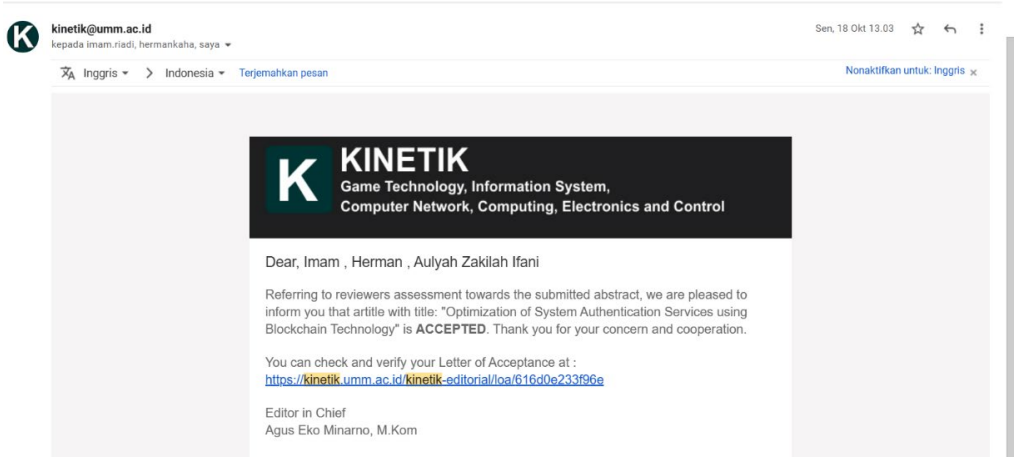
Sudiarto Raharjo, W., E.K. Ratri, I. D., & Susilo, H. (2017). Implementasi Two Factor Authentication Dan Protokol Zero Knowledge Proof Pada Sistem Login. *Jurnal Teknik Informatika Dan Sistem Informasi*, *3*(1), 127–136. https://doi.org/10.28932/jutisi.v3i1.579

T, G. S., & Sasikala, D. (2019). Vulnerability Assessment of Web Applications using Penetration Testing. *International Journal of Recent Technology and Engineering*, *8*(4), 1552–1556. https://doi.org/10.35940/ijrte.B2133.118419

Wibowo, F., Harjono, H., & Wicaksono, A. P. (2019). Uji Vulnerability pada Website Jurnal Ilmiah Universitas Muhammadiyah Purwokerto Menggunakan OpenVAS dan Acunetix WVS. *Jurnal Informatika*, *6*(2), 212–217. https://doi.org/10.31311/ji.v6i2.5925

b. Jurnal Kinetik





Optimization of System Authentication Services using Blockchain Technology

Imam Riadi[1], Herman[2] ,Aulyah Zakilah Ifani[3*]
1, Departement of Information System[1]
2, Departement of Informatics[2,3]
3, Universitas Ahmad Dahlan, Indonesia[1,2,3]

Article Info

Keywords:
Authentication, *Blockchain*, *Networkminer*, NFLDC, *Wireshark*.

Article history:

Cite:

* Corresponding author.
Aulyah Zakilah Ifani
E-mail address:
aulyah1908048022@webmail.uad.ac.id

Abstract

The development of the era, one thing that must be considered for security is the Login System. In most cases, user login information is stored on the server. This gives access to sensitive information, many hackers easily break into data from users.Based on these problems, this research focuses on data security authentication in the form of usernames and passwords in the login system. Authentication using blockchain is used to reduce malicious access and increase security for the authentication process. One of the innovation technologies that can solve these problems is Blockchain Technology. Using blockchain technology, hackers will find it difficult to change and modify the same data on all computers at the same time because it takes a very long time to crack the encryption code on each block of data in the entire computer network. Data storage or transactions in the blockchain are stored in the form of hashes. This makes it difficult for hackers to break into it. Tests in this study using wireshark tools and networkminer. Based on the research conducted, the test was conducted as many as 5 times with two scenarios, namely authentication of the login system before using the blockchain and after using the blockchain. The results obtained The system built using blockchain is able to secure data. The test results obtained that data in the form of usernames and passwords were converted into hashes and with the immutable nature of the blockchain, data from users could not be changed or replaced by anyone.

1. Introduction

Nowadays the internet is becoming very famous, the use of the internet is not only among young people but among adults many who use the internet make it a necessity. It is this need of the internet that makes it very easy for a person to access it[1]. Services to users who need a wireless network can be accessed through smartphones, laptops, and other mobile devices[2][3]. The advantages obtained in using wireless in terms of mobility also get a big challenge in the form of securing the wireless network, many illegal attacks to get usernames and passwords from users (Kunang & Ibadi, 2013). Security systems and login processes usually use usernames and passwords as authentication methods. It is used because of the ease of implementing it (Putra et al., 2018). Disadvantages when using authentication are usernames and passwords that are stored in a database and provide an easy effect to hack for example using wireshark and network miner [5]. Authentica

tion is a proof of an entity's identity. For example credit cards, or machines and people [2].

Blockhain is a decentralized and distributed technology [7][8]. Blockchain is defined as a distributed database [9]. The important thing about blockchain has three elements: repicated ledger, cryptography, peer-to-peer networking [10]. Blockhain is an additional security while cryptography maintains the confidentiality and authentication of data exchange[11]. Blockchain technology makes it difficult for hackers to change and modify the same data on all computers at the same time because it takes a very long time to decode the encryption on every block of data across the computer network. There are two types of records on a blockchain system: transactions and blocks. Blockchain transactions are stored in one block together. Each block forms a network that contains cryptographic algorithms. Cryptographic algorithms are used to retrieve data from previous blocks and convert them into compact strings that can detect sabotage (Fauzan N I, 2018)(Wan et al., 2019). Each block has a hash value and each of those blocks gets a value from the previous hash [14]. Figure 1 shows the blockchain scheme.

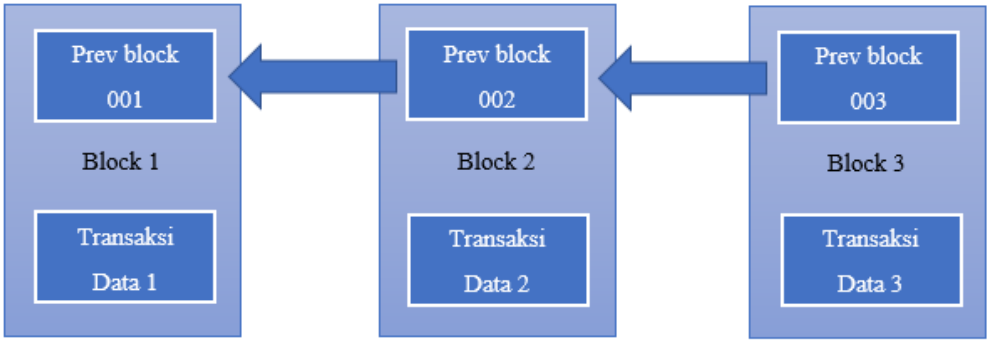


Figure 1. Blockchain Scheme

Figure 1 describes each block connected through a hash of the previous block. The first block has a connection with the previous block by entering hash and proof values. Each new block has an attribute called a hashid which is the hash (SHA256) of 4 strings i.e. hash id, index, data, and timestamp taken from the previous block. The initial node of the blockchain is called the genesis block and the value of the previous block is given a value of zero. Timestamp to record node time (Deep et al., 2019).

The workings of the hash function is that the length of the variable is converted into a binary form but with a fixed length. Hash functions are used in security applications, such as message authentication, password storage and digital signatures[16]. Blockchain was deliberately created to change transactions A and B occur without intermediaries, the cost is much cheaper compared to other transactions and blockchain is also much more secure (Rahardja et al., 2020)(Linoy et al., 2019). Blockchain is a tamper-proof block stored in each participating node. Each block records a set of related metadata data transactions [19]. Blockchain contains transactions stored in a block of data[14]. Each block stores a hash derived from the previous block so that a block chain is formed. The first block is called the genesis block which is the only block that does not have a hash of the previous block[20][21]. Blockchains also have cryptographic algorithms that keep strands of the blocks secured in the blockchain[22]. This algorithm will make it easier for the system to track in case of sabotage of the block chain (Fauzan N I, 2018) (Singh et al., 2021).

Some previous research related to blockchain technology as an adjunct security is as follows. Research from Fat et al., on securitizing sensor data on IoT networks. The results of this study suggest that the Internet of Things can use blockchain for securitization of data (Fat et al., 2019). Zhang et al., conducted research on blockchain security and orivation. The result is that developing cryptographic algorithms as well as other security and privacy methods will be key to enabling the technology in the development of blockchain and its applications in the future. Blockchain provides security and cryptography as authentication (Zhang et al., 2019). Singla et al., conducted research by developing a leave application using smart contracts. The system in this study was successfully developed using solidity as well as ethereum (Singla et al., 2019). Aprialim et al., research on the application of blockchain with the integration of smart contracts on crowfounding systems that explain crowfunding systems operate using a centralized system, so as not to provide data security and transparency of fundraising activities in full. The functional needs of the system are proven to work according to the design of the use case for either fundraising users or funders [24]. Hattab &

Taha., research on blockchain cryptocurrency technology (crypto currency), aims to illustrate the opportunities obtained from the technology that plays a role in cryptocurrencies (Noorsanti et al., 2018). Rahardja et al., This research shows that blockchain can be used as an easier, more efficient payment, and the blockchain makes the system as tracking, and can synchronize the data in the blockhain to all users (Rahardja et al., 2020). Fadlil et al., Explaining that by modifying cryptocurrencies and blockchain technology can function properly on the system, it can be proven by the success of chain validation that detects the or absence of immutability data displayed in valid columns that are true or false (Fadlil et al., 2020). Next about security and privacy on blockchain [26]. Research on data security for school service top-up transactions combination blockchain technology modification. The results obtained from this research are that by modifying cryptography and blockchain technology can function properly on the system[27]. Various applications of blockchain technology are about health, Internet of Things (IoT), fundraising, digital asset management, education, etc [28][29][30]. Blockchain is also used to enhance the security of e-commerce systems[31] as well as e-voting systems[32]. Blockchain can give confidence to third parties who oversee the process between sellers and buyers to confirm the authenticity of data and information (Shorman et al., 2019).

Based on previous research this research aims to secure the login security system with blockchain authentication and conduct system testing. Blockhain uses the digital signature feature to make transactions so that data from users cannot be changed and modified. This is the advantage of blockchain. This blockchain-based authentication system can be used to secure authentication processes in a decentralized and irreversible way. The login security system is modified using blockchain technology. Comparison with previous research as literature to get new research can be found several aspects that this research uses the NFLDC framework as a process flow of this research, for security optimization using blockchain technology, and based on the tools used to build blockchain need to use several tools such as metamask, ganache, truffle, solidity, web3js, nodejs, reacts. As for system testing using wireshark and network miner

2. Research Method

The Login System is an application used as an object in this study. This application is an application or website that will be used in various other applications. The login system has a username and password that will be filled in first before logging into the system. Entering usernames and passwords becomes very vulnerable to hacking. The number of tools available makes user data threatened. This will be very dangerous if left continuously. Therefore, in this research will be done optimization of login system security. Optimization of this system using blockchain is expected to be used as a security of the login system. The test was conducted using two tools, wireshark and network miner. Wireshark analyzes network packets in as much detail as possible (Richard Sharpe, 2014). *Wireshark uses pcap to capture packets, .pcap files, IP sources and destinations, protocols involved including protocol header data[35]. This is particularly useful for this study because it can evaluate security breach events to solve network security issues[36]. This research wireshark can take all the packets that pass and select them in as much detail as possible, for example username and password. As for networkminer packet analizer is used as a sniffer or tool to capture packet data that detects the operating system, hostname, session, open ports, etc. without charging a trace of traffic on the network. Network miners can parse .pcap files for off-line analysis [37]. Tested for weak usernames and passwords on the system, especially using Wireshark.* After getting the results, then the results are transferred to Network Miner. This is done to get results from Network Miner.

The research phase as a process flow of this research is guided by the Network Forensic Development Life Circle (NFLDC) framework. The NFLDC is a combination of the Information System Development Life Circle (ISDLC) and Network Forensic Readiness (NFR) methods. The concept of NFR maximizes the ability to gather credible evidence while minimizing the cost of inside response. This is a recommendation to increase the efficiency of investigation. But there is little to discuss how to integrate NFR into a network of systems. NFR appears to investigate malicious online intruders. In this case NFR as a case study. ISDLC is designed to incorporate security throughout the system development cycle. The ISDLC methods of each phase are analyzed and modified to include additional steps that create digital forensic embedding. Specific ISDLC modifications result in nfldc. The NFLDC research phase contains initiation,

acquisition, implemengtation, operations and disposition. Figure 3 shows the system development process with the NFLDC.

Figure 2. NFLDC Methodology

The NFLDC method as figure 2 generally describes the five main stages of performing a forensic process. The stages used include initiation, acquisition, implementation, and also operation. The NFLDC method was chosen because it has a practical and easy to implement research phase and has integration about the findings obtained in the forensic process.

3. Results and Discussion

The login system on this research becomes the main object in the vulnerability testing stage using Blockchain Technology. This research requires software and hardware to develop the system. Table 1 is a tool and material needed in the research process.

Table 1. Research Tools and Materials

Tools and Materials	Information	Type
Laptop	Tester	Hardware
Network Miner	Penetration Testing	Software
Wireshark	Penetration Testing	Software
Metamask	Wallet Ethereum	Software
Visual Studio Code	Programming	Software
Ganache	Languange	Software
Truffle	Blockchain Emulator Framework	Software

Table 1 as a tool and material needed in this study used to facilitate the process of research. Lenovo laptop as tester to perform scans in this study. Truffle as a development framework for daaps (decentralized applications) based on the ethereum blockchain. Ganache is part of the trufflesuite, which is used to run local ethereum servers. Solidity is used as a contract-oriented programming language for writing smart contracts. Metamask is used as a bridge that allows users to go to a web browser. Rects as a javascript library for building user interfaces.

3.1 Initiation

Case simulation of the system using the Visual Studio Code programming language. The result of the simulation stage is a system that is ready to be tested and run. The login system utilizes the Ethereum blockchain platform which implements blockchain technology and smart contracts. Web3js as an application programming interface to connect browsers with an extension called metamask as a bridge between the login system and the ethereum blockchain. This metamask acts as an ethereum wallet for information management. Meanwhile, smart contracts are built using the visual programming language studio code. This stage of research the user must be a member of the blockchain network, when he becomes a member then the user will have an ethereum account. An ethereum blockchain-based application must run a smart contract. So for smart contracts will be signed first by people who already have an ethereum account to run an Ethereum-based application. The user is next if you want to login then must have registered as a smart contract signer. Login data from users will be stored as a hash to the blockchain via smart contract. The user requests access to the website, for example a hash derived from the user's granted credentials compared to the hash stored in the Smart Contract. So, when the user logins and the user data is suitable then the user is authorized to access the web. Conversely, if it does not fit then access is denied. Each user is required to

connect with an Ethereum address that was previously done in the registration process, as this address generates the user's login hash.

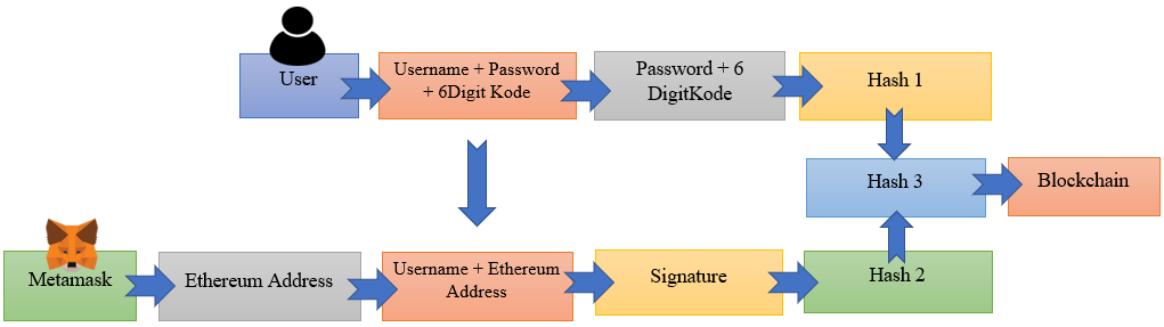


Figure 3. Steps to Generate a Hash of User Data

Figure 3 shows the steps to generate a hash of user data in the form of username, password, 6-digit code, ethereum address. When the user registers, the user must first fill out a form to provide a username, password, 6-digit code and ethereum address. Ethereum's address is taken from a wallet (ganache). Ethereum addresses are linked to usernames to generate signatures through the web3 function. The resulting signature is hash (hash 1). Furthermore, a password connected with a 6 digit code is used to generate another hash (hash 2). Then the two hashes are combined to produce the final result that is stored into a smart contract. Users who want to enter the login system must be connected to the blockchain with the same address registered when registering. After filling in the username, password, 6 digit code correctly, then the back end code generates a hash with the user's login information and compares it to the hash stored in the smart contract by the ethereum address that asks to login. If the two hashes match then the user successfully login.

3.2 Acquisition

The system that is built will be tested under two conditions, namely the system before implementation and the system after implementation using blockchain technology. System before implementation, when registering and doing the login process, users will directly enter the system without going through the authentication process. So that the process is vulnerable to break-ins. Unlike the conditions after using the blockchain, the system will perform the authentication process first the user haris becomes a member on the blockchain network. When the user becomes a member, the user will have an ethereum id. Simulation of blockhain using ganache. The system after using blockchain must run a smart contract. So that the smart contract will be signed by users who have ether id. The user if you want to login must have registered as a smart contract signer.

3.3 Implementation

The Login system in this study utilizes the use of the Ethereum blockchain platform that implements blockchain technology and smart contracts. Web3js as an application programming interface to connect browsers with an extension called metamask as a bridge between the login system and the ethereum blockchain. This metamask acts as an ethereum wallet for information management. Meanwhile, smart contracts are built using a solidity programming language. This research user must be a member of the blockchain network. First the user creates an account before accessing the network. Ganache becomes the place to get networked. An ethereum blockchain-based application must run a smart contract. Smart contracts will be signed first by people who already have an Ether ID to run an Ethereum-based application. Ganache already provides 10 default accounts and each account has a balance of 100 eth. The account is used for transactions on the ganache blockchain.

The user interface of ganache has 10 ethereum addresses that have 100 eth that can be used for blockchain simulation. This one ethereum network is used for one registration processSo if the user wants to register a new account to enter the login system then the user takes one ethereum network that is on the ganache. The user if you want to login must have registered as a signer of the smart contract. Before the user logs into the system page, the user first connects the ganache to metamask. This metamask as a bridge

for users to log into the web browser. This metamask allows users to run ethereum daaps directly in the browser. Metamask account that is already connected to the ethereum network, after which the user then performs the registration process in order to log in to the system. After the user registers, then the transaction details will appear to continue the transaction that has been successfully shown in Figure 4.

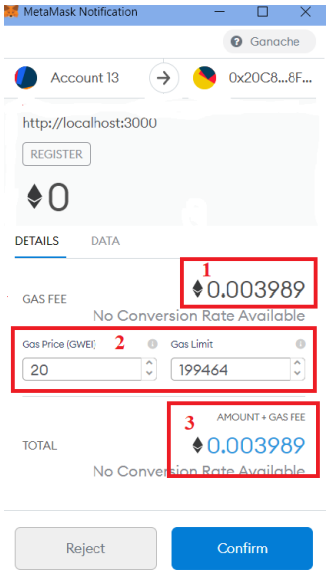


Figure 4. Transaction Details

Figure 4 shows the transaction details, where the Gas Fee (box 1) is the transaction fee paid at 0.003989. Gas Price (box 2) is paid for 20 gwei which means the transaction will be processed quite quickly. While the Gas Limit (box 2) is 199464. Amount + Gas Fee is the amount to be issued to make a transaction (box 3). When the user initiates the transaction, the request will go to the ethereum blockchain where the miner made the transaction. The price to be paid is gas limit and gas price. Once the transaction is successful it will be recorded on the blockchain and users can see the gas used. When the transaction is declared successful, the user can login to the system using a username, password, 6 digit code that has been registered.

System testing is done with two scenarios: the login system before using the blockchain and after using the blockchain and using wireshark tools and network miner. Users will login by filling in a username and password. When the user logins the attacker also captures the package by using wireshark to be able to get http protocol files. Wireshark will stamp the captured packages. To get the username and password file then filter the package with the HTTP protocol file. Before using blockchain technology, users will login by entering a username and password then the attacker will easily get data from the user. This is because the login system has not been secured. Here's figure 5 that shows user data.

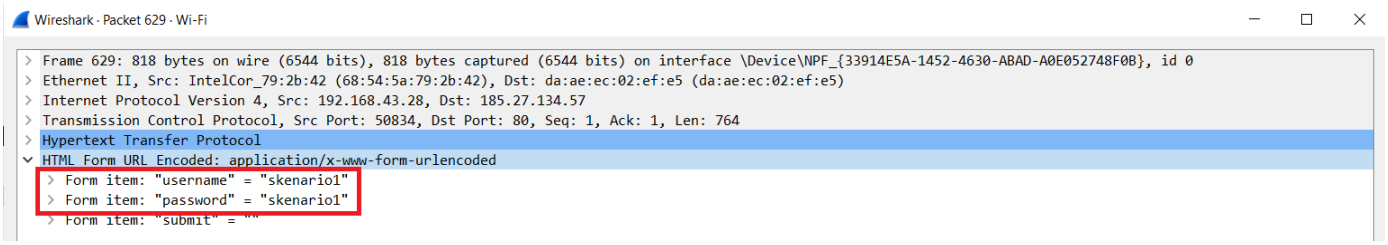


Figure 5. File Showing User Data

Figure 5 above uses a login system that has not been secured, so that the security of user data transmitted over the network will be easily detected by attackers (Figure 6). So, to secure it on this research using blockhain technology that can secure data in the form of usernames and passwords from users. Testing of the login system after using the blockchain can be seen in figure 6.

```
issuerNameHash: c72e798addff6134b3baed4742b8bbc6c0240763
issuerKeyHash: 8a747faf85cdee95cd3d9cd0e24614f371351d27
serialNumber: 0x01675c9687a505fc0a00000000f6eacb
```

Figure 6. Filter HTTP Packets After Using Blockchain

Figure 6 shows the results of http packet filtering after using blockchain. Furthermore, in http data there is information such as IP address 77.214.45.70 source and 192.168.1.7 destination. If the login data has been added blockchain technology then the result of the analysis of http capture packets from wireshark is data that has been converted into hash form so that for attackers can not see the username and password sent. After testing using wireshark, the second test will be done using a networkminer. The data results are the same as those that previously used the login system before and after using blockchain technology. Files from wireshark are stored using the format (.pcap) after which testing will be performed. Before using blockchain data in the form of usernames and passwords can be read by attackers (figure 5). Figure 7 shows the capture before using the blockchain.

Client	Server	Protocol	Username	Password	Valid login	Login timestamp
192.168.43.28 (Windows)	185.27.134.57 [aulyah.epizy.com]	HTTP Cookie	PHPSESSID=f04239a5190b6b201785fcc561b33593; t	N/A	Unknown	2021-08-19 15:50:31 UTC
192.168.43.28 (Windows)	185.27.134.57 [aulyah.epizy.com]	MIME/MultiPart	skenario1	skenario1	Unknown	2021-08-19 15:51:02 UTC

Figure 7. Capture Results Before Using Blockchain

Figure 7 generates data from the user in the form of a username and password. Unlike the login system that uses blockchain, data from users cannot be read by attackers. This is because the data has been converted into a hash form that makes it difficult for attackers to get usernames and passwords (figure 6). The capture results use the networkminer in figure 8.

gts1c3[6].ocsp-request - File Details

Name	gts1c3[6].ocsp-request
MD5	b650ea3645acc57b6c6f546bcd99cd32
SHA1	7de1c31a22280751e393b44ceaa6dfc7831f2522
SHA256	94f74aae83e5d54ac922124ab653c4bdeee21e1b5c85ca2a87852ef613774f69
Path	C:\Users\ASUS\Downloads\NetworkMiner_2-7-1\AssembledFiles\192.168.10.244\TCP-49681\gts1c3[6].ocsp-request
Size	83
LastWriteTime	16/08/2021 13:58
Source	192.168.10.244 [LAPTOP-17T3APHH] (Windows)
Destination	172.217.194.94 [pki-goog.l.google.com] [ocsp.pki.goog]

Max bytes to read: 256 Font size: 10

```
3051304F304D304B3049300906052B0E      0Q0O0M0K0I0...+.
03021A05000414C72E798ADDFF6134B3      .......?.y???a4?
BAED4742B8BBC6C024076304148A747F      ??GB????$.c..?t.
AF85CDEE95CD3D9CD0E24614F371351D      ??????=???F.?q5.
27021001675C9687A505FC0A00000000      '...g\???.?.....
F6EACB                                ???
```

Figure 8. Capture Results After Using Blockchain

Figure 8 shows the results after using the blockchain. User data has been changed to hash form. Attackers do not get data from users.

3.4 Operation/Maintance

After doing some testing obtained results using wireshark as in table 2. testing before implementing blockchain and after implementing blockchain

Table 2. Test Results with Tools

Wireshark	Before blockchain implementation	Username: skenario1 Password: skenario1
	After blockchain implementation	c72e798addff6134b3baed4742b8bbc6c02407638a747faf85 cdee95cd3d9cd0e24614f371351d2701675c9687a505fc0a 00000000f6eacb
Networkminer	Before blockchain implementation	Username: skenario1 Password: skenario1
	After blockchain implementation	3051304f304d304b3049300906052b0e03021a05000414c72 e798addff6134b3baed4742b8bbc6c024076304148a747faf85 cdee95cd3d9cd0e24614f371351d27021001675c9687a505fc 0a00000000f6eacb

Table 2 of the tests that have been conducted shows the conditions before and after blockchain technology is implemented. Prior to implementation, the system displayed vulnerabilities to steal data in the form of usernames and passwords that could be detected using wireshark and networkminer tools. After implementing blockchain technology, the transmitted data becomes secure because it is converted into hash form. Table 3 shows the comparison between the proposed scheme and the previous authentication scheme.

Table 3. Comparison of the security of the proposed scheme

Attack	System Authentication Blockchain	Shajina and Varalakhsami (Shajina & Varalakshmi, 2017)	Yang et all (Yang et al., 2019)	Anakath et all (Anakath et al., 2019)
Password Guessing Attack	Yes	No	No	Yes
Username Guessing Attack	Yes	No	No	No
Prevent replay Attack	Yes	Yes	Yes	Yes
Prevent Insider Attack	Yes	No	Yes	No
Prever Impersonation Attack	Yes	Yes	No	Yes

Table 3 shows solutions that guarantee key security requirements. Blockchain authentication is very efficient in its operation.

3.5 Disposition

Disposition is the part where the data storage process is carried out. In this case, the data in question is the testing data from the login system. One relevant form of implementation applied to support documentation is to use an external metada .pcap approach to store wireshark files, so that data is safe for users. The disposition stage shows that blockchain technology can be applied to login authentication systems in securing user data. This is evidenced by the entire data from the user in the form of a username, password has been encrypted or converted into a hash. The tests showed that the overall test results of the login authentication system before and after using the blockchain using two tools namely wireshark and networkminer showed the same results. The results are obtained based on testing the login authentication system using tools, scenarios, and samples that have been determined at the initiation stage.

4. Conclusion

Research on this login system application uses figma in the creation of its design and system implementation using visual studio code. Blockchain technology has been successfully used to improve the security of the login system, with user login data stored as hashes to the blockchain via smart contracts. Tests were conducted that showed the condition of the system before using the blockchain and after using the blockchain. The results obtained by the system using blockchain technology managed to secure data in the form of usernames and passwords by converting the data into hashes. So for attackers who want to get user data has difficulty. Systems built using blockchain are able to secure data. This is evidenced by testing using wireshark and networkminer tools. The results of the test obtained data in the form of usernames and passwords converted into hashes and with immutable blockchain properties so that data from users cannot be changed or replaced by anyone.

References

Abdillah, M. A., Yudhana, A., & Fadil, A. (2020). Sniffing Pada Jaringan WiFi Berbasis Protokol 802.1x Menggunakan Aplikasi Wireshark. *J-SAKTI (Jurnal Sains Komputer Dan Informatika)*, *4*(1), 1. https://doi.org/10.30645/j-sakti.v4i1.181

Alam, A., Zia Ur Rashid, S. M., Abdus Salam, M., & Islam, A. (2018). Towards Blockchain-Based E-voting System. *2018 International Conference on Innovations in Science, Engineering and Technology, ICISET 2018*, 351–354. https://doi.org/10.1109/ICISET.2018.8745613

Anakath, A. S., Rajakumar, S., & Ambika, S. (2019). Privacy preserving multi factor authentication using trust management. *Cluster Computing*, *22*, 10817–10823. https://doi.org/10.1007/s10586-017-1181-0

Aprialim, F., Adnan, & Paundu, A. W. (2021). Penerapan Blockchain dengan Integrasi Smart Contract pada Sistem Crowdfunding. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *5*(1), 148–154. https://doi.org/10.29207/resti.v5i1.2613

Arse, M., & Dubey, J. (2020). A Survey of Internet of Things node's transactions Secure through Blockchain Technology. *International Journal of Computer Applications*, *175*(25), 33–37. https://doi.org/10.5120/ijca2020920796

Bragagnolo, S., Rocha, H., Denker, M., & Ducasse, S. (2018). SmartInspect: Solidity smart contract inspector. *2018 IEEE 1st International Workshop on Blockchain Oriented Software Engineering, IWBOSE 2018 - Proceedings*, *2018-Janua*, 9–18. https://doi.org/10.1109/IWBOSE.2018.8327566

Chaniago, N., Sukarno, P., & Wardana, A. A. (2021). Electronic document authenticity verification of diploma and transcript using smart contract on ethereum blockchain. *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, *7*(2), 149–163. https://doi.org/10.26594/REGISTER.V7I2.1959

Cui, Y., Cui, J., & Hu, J. (2020). A Survey on XSS Attack Detection and Prevention in Web Applications. *ACM International Conference Proceeding Series*, 443–449. https://doi.org/10.1145/3383972.3384027

Damai, S., Hu, K., Palit, H. N., Handojo, A., Studi, P., Informatika, T., Industri, F. T., Petra, U. K., & Surabaya, J. S. (2019). Implementasi Blockchain : Studi Kasus e-Voting. *Jurnal Infra Petra*, *031*.

Deep, G., Mohana, R., Nayyar, A., Sanjeevikumar, P., & Hossain, E. (2019). Authentication protocol for cloud databases using blockchain mechanism. *Sensors (Switzerland)*, *19*(20), 1–13. https://doi.org/10.3390/s19204444

Dhammearatchi, D. (2015). *U SE O F N ETWORK F ORENSIC M ECHANISMS*. *7*(4), 21–36.

Efanov, D., & Roschin, P. (2018). The all-pervasiveness of the blockchain technology. *Procedia Computer Science*, *123*, 116–121. https://doi.org/10.1016/j.procs.2018.01.019

El-Sofany, H. F. (2020). A new cybersecurity approach for protecting cloud services against DDoS attacks. *International Journal of Intelligent Engineering and Systems*, *13*(2), 205–215. https://doi.org/10.22266/ijies2020.0430.20

Endicott-Popovsky, B. E., & Frincke, D. A. (2006). Embedding forensic capabilities into networks: Addressing inefficiencies in digital forensics investigations. *Proceedings of the 2006 IEEE Workshop on Information Assurance*, *2006*, 133–139. https://doi.org/10.1109/iaw.2006.1652087

Endicott-Popovsky, B., Frincke, D. A., & Taylor, C. A. (2007). A theoretical framework for organizational network forensic readiness. *Journal of Computers*, *2*(3), 1–11. https://doi.org/10.4304/jcp.2.3.1-11

Fadlil, A., Riadi, I., & Nugrahantoro, A. (2020). Data Security for School Service Top-Up Transactions Based on AES Combination Blockchain Technology. *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, *11*(3), 155. https://doi.org/10.24843/lkjiti.2020.v11.i03.p04

Fat, J., Candra, H., & Wiliam, W. (2019). Sekuritisasi Data Sensor Pada Aplikasi Internet of Things (IoT) Dengan Menggunakan Blockchain Ethereum Di Jaringan Testnet. *TESLA: Jurnal Teknik Elektro*, *21*(1), 79. https://doi.org/10.24912/tesla.v21i1.5886

Fauzan N I, A. (2018). *TEKNOLOGI BLOCKCHAIN DAN PERANANNYA DALAM ERA DIGITAL*. *4*, 1–15.

Gitanjali Simran T, S. D. (2019). *Vulnerability Assessment of Web Applications using Penetration Testing*. *4*, 1552–1556. https://doi.org/10.35940/ijrte.B2133.118419

Gupta, Shashank, & Gupta, B. B. (2017). Cross-Site Scripting (XSS) attacks and defense mechanisms: classification and state-of-the-art. *International Journal of Systems Assurance Engineering and Management*, *8*, 512–530. https://doi.org/10.1007/s13198-015-0376-0

Gupta, Suyash, & Sadoghi, M. (2020a). Encyclopedia of Big Data Technologies. *Encyclopedia of Big Data Technologies*, *May*. https://doi.org/10.1007/978-3-319-63962-8

Gupta, Suyash, & Sadoghi, M. (2020b). Encyclopedia of Big Data Technologies. *Blockchain Transaction Processing*, *May*. https://doi.org/10.1007/978-3-319-63962-8

Harahap, E. P., Aini, Q., & Anam, R. K. (2019). Pemanfaatan Teknologi Blockchain Pada Platform Crowdfunding. *Technomedia Journal*, *4*(2), 199–210. https://doi.org/10.33050/tmj.v4i2.1108

Hidayat, T. N., & Riadi, I. (2021). Optimation Wireless Security IEEE 802.1X using the Extensible Authentication Protocol-Protected Extensible Authentication Protocol (EAP-PEAP). *International Journal of Computer Applications*, *174*(11), 25–30. https://doi.org/10.5120/ijca2021920988

Ismanto, L., Ar, H. S., Fajar, A. N., Sfenrianto, & Bachtiar, S. (2019). Blockchain as E-Commerce Platform in Indonesia. *Journal of Physics: Conference Series*, *1179*(1). https://doi.org/10.1088/1742-6596/1179/1/012114

Khera, Y., Kumar, D., Sujay, S., & Garg, N. (2019). Analysis and Impact of Vulnerability Assessment and Penetration Testing. *Proceedings of the*

International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon 2019, 525–530. https://doi.org/10.1109/COMITCon.2019.8862224

Kiran, K. V. V. N. L. S., Devisetty, R. N. K., Kalyan, N. P., & Mukundini, K. (2020). ScienceDirect ScienceDirect ScienceDirect Building a Intrusion Detection System for IoT Environment using Machine Learning Techniques Building a Intrusion Detection System for IoT Environment using Machine Learning Techniques. *Procedia Computer Science*, *171*(2019), 2372–2379. https://doi.org/10.1016/j.procs.2020.04.257

Kunang, Y. N., & Ibadi, T. (2013). Celah Keamanan Sistem Autentikasi Wireless Berbasis RADIUS. *Celah Keamanan Sistem Autentikasi Wireless Berbasis RADIUS*, *34*(2), 1907–5022. https://doi.org/DOI:10.13140/RG.2.1.2115.0323

Linoy, S., Mahdikhani, H., Ray, S., Lu, R., Stakhanova, N., & Ghorbani, A. (2019). Scalable privacy-preserving query processing over ethereum blockchain. *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, 398–404. https://doi.org/10.1109/Blockchain.2019.00061

Marques, N., Zúquete, A., & Barraca, J. P. (2019). *Integration of the Captive Portal paradigm with the 802.1X architecture*. 1–28. http://arxiv.org/abs/1908.09927

Mohanta, B. K., & Jena, D. (2018). An Overview of Smart Contract and Use Cases in Blockchain Technology: 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT). *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 1–4.

Mokbal, F. M. M., Dan, W., Imran, A., Jiuchuan, L., Akhtar, F., & Xiaoxi, W. (2019). MLPXSS: An Integrated XSS-Based Attack Detection Scheme in Web Applications Using Multilayer Perceptron Technique. *IEEE Access*, *7*, 100567–100580. https://doi.org/10.1109/ACCESS.2019.2927417

Noorsanti, R. C., Yulianton, H., & Hadiono, K. (2018). Blockchain - Teknologi Mata Uang Kripto (Crypto Currency). *Prosiding SENDI_U*, *3*(November), 306.

Nurfaizi, M. C., Bhawiyuga, A., & Amron, K. (2019). *Pengembangan Gateway untuk Menghubungkan Jaringan IoT (Internet Of Things) Dan Jaringan Blockchain*. *3*(12), 10949–10958.

Pallavi, C., Girija, R., & Jayalakshmi, S. L. (2021). An Analysis on Network Security Tools and Systems. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3833455

Patel, K. (2019). A survey on vulnerability assessment penetration testing for secure communication. *Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019*, *Icoei*, 320–325. https://doi.org/10.1109/ICOEI.2019.8862767

Putra, A. W. P., Bhawiyuga, A., & Data, M. (2018). Implementasi Autentikasi JSON Web Token (JWT) Sebagai Mekanisme Autentikasi Protokol MQTT Pada Perangkat NodeMCU. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *2*(2), 584–593. http://j-ptiik.ub.ac.id

Rahardja, U., Aini, Q., Yusup, M., & Edliyanti, A. (2020). Penerapan Teknologi Blockchain Sebagai Media Pengamanan Proses Transaksi E-Commerce. *Computer Engineering, Science and System Journal*, *5*(1), 28–32. https://doi.org/10.24114/CESS.V5I1.14893

Rahardja, U., Harahap, E. P., & Christianto, D. D. (2019). Pengaruh Teknologi Blockchain Terhadap Tingkat Keaslian Ijazah. *Technomedia Journal*, *4*(2), 211–222. https://doi.org/10.33050/tmj.v4i2.1107

Riadi, I., Umar, R., Aziz, M. A., Informatika, S. T., & Dahlan, U. A. (2021). *Komparatif Web-based Instant Messaging Vulnerability Menggunakan*. *1*(10), 813–819.

Riadi, I., Umar, R., & Busthomi, I. (2020). *Optimasi Keamanan Autentikasi dari Man in the Middle Attack (MiTM) Menggunakan Teknologi Blockchain*. *04*, 15–19.

Riadi, I., Umar, R., & Lestari, T. (2020). Analisis Kerentanan Serangan Cross Site Scripting (XSS) pada Aplikasi Smart Payment Menggunakan Framework OWASP. *JISKA (Jurnal Informatika Sunan Kalijaga)*, *5*(3), 146. https://doi.org/10.14421/jiska.2020.53-02

Riadi, I., Umar, R., & Nasrulloh, I. M. (2018). Analisis Forensik Digital Pada Frozen Solid State Drive Dengan Metode National Institute of Justice (Nij). *Elinvo (Electronics, Informatics, and Vocational Education)*, *3*(1), 70–82. https://doi.org/10.21831/elinvo.v3i1.19308

Richard Sharpe, E. W. (2014). *Wireshark User ' s Guide*. 191.

Rizky, A., Kurniawan, S., Gumelar, R. D., Kurniawan, V., & Prakoso, M. B. (2021). Use Of blockchain technology in implementing information system security on education. *BEST (Journal of Biology Education Sains & Technology)*, *4*(1), 62–70.

Rusdan, M., & Sabar, M. (2020). Analisis dan Perancangan Jaringan Wireless Dengan Wireless Distribution System Menggunakan User Authentication Berbasis Multi-Factor Authentication. *JOINT (Journal of Information Technology)*, *02*(01), 17–24.

Salman, T., Zolanvari, M., Erbad, A., Jain, R., & Samaka, M. (2019). Security services using blockchains: A state of the art survey. *IEEE Communications Surveys and Tutorials*, *21*(1), 858–880. https://doi.org/10.1109/COMST.2018.2863956

Sarmah, U., Bhattacharyya, D. K., & Kalita, J. K. (2018). A survey of detection methods for XSS attacks. *Journal of Network and Computer Applications*, *118*, 113–143. https://doi.org/10.1016/j.jnca.2018.06.004

Shajina, A. R., & Varalakshmi, P. (2017). A novel dual authentication protocol (DAP) for multi-owners in cloud computing. *Cluster Computing*, *20*(1), 507–523. https://doi.org/10.1007/s10586-017-0774-y

Shorman, S., Allaymoun, M., & Hamid, O. (2019). Developing the E-Commerce Model a Consumer To Consumer Using Blockchain Network Technique. *International Journal of Managing Information Technology*, *11*(02), 55–64. https://doi.org/10.5121/ijmit.2019.11204

Sikos, L. F. (2018). AI in Cybersecurity. In *Springer*. https://doi.org/10.1007/978-3-319-98842-9

Singh, S., Sanwar Hosen, A. S. M., & Yoon, B. (2021). Blockchain Security Attacks, Challenges, and Solutions for the Future Distributed IoT Network. *IEEE Access*, *9*, 13938–13959. https://doi.org/10.1109/ACCESS.2021.3051602

Singla, V., Malav, I. K., Kaur, J., & Kalra, S. (2019). Develop Leave Application using Blockchain Smart Contract. *2019 11th International Conference on Communication Systems and Networks, COMSNETS 2019*, *2061*, 547–549. https://doi.org/10.1109/COMSNETS.2019.8711422

Subekti, Z. M., & Subandri, S. (2020). Implementasi Metode Per Connection Queue Dengan Access User Direct Mac Filtering Pada Jaringan Wireless. *INOVTEK Polbeng - Seri Informatika*, *5*(2), 240. https://doi.org/10.35314/isi.v5i2.1472

Susianto, D., Rachmawati, A., Informatika, J. M., Minner, N., Cendikia, C., Lampung, B., & Industri, F. T. (2018). *IMPLEMENTASI DAN ANALISIS JARINGAN MENGGUNAKAN WIRESHARK , CAIN AND ABELS , NETWORK MINNER (Studi Kasus : AMIK Dian Cipta Cendikia)*. *XVI*, 120–125.

Teferi, F., & Nixon, J. S. (2019). A Security Mechanism to Mitigate DDoS Attack on Wireless Local Area Network (WLAN) using MAC with SSID. *International Journal of Computer Sciences and Engineering*, *7*(4), 864–869. https://doi.org/10.26438/ijcse/v7i4.864869

Tian, Y., Zheng, N., Chen, X., & Gao, L. (2021). Wasserstein Metric-Based Location Spoofing Attack Detection in WiFi Positioning Systems. *Security and Communication Networks*, *2021*. https://doi.org/10.1155/2021/8817569

Toapanta, S. M., Escalante Quimis, O. A., Mafla Gallegos, L. E., & Maciel Arellano, M. R. (2020). Analysis for the evaluation and security management of a database in a public organization to mitigate cyber attacks. *IEEE Access*, *8*, 169367–169384. https://doi.org/10.1109/ACCESS.2020.3022746

Umar, R., Riadi, I., & Zamroni, G. M. (2018). Mobile forensic tools evaluation for digital crime investigation. *International Journal on Advanced Science, Engineering and Information Technology*, *8*(3), 949–955. https://doi.org/10.18517/ijaseit.8.3.3591

Vimala, S. T., & Dhas, J. P. M. (2018). SDN based DDoS attack detection system by exploiting ensemble classification for cloud computing. *International Journal of Intelligent Engineering and Systems*, *11*(6), 282–291. https://doi.org/10.22266/IJIES2018.1231.28

Wan, L., Eyers, D., & Zhang, H. (2019). Evaluating the impact of network latency on the safety of blockchain transactions. *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, 194–201. https://doi.org/10.1109/Blockchain.2019.00033

Wang, S., Yuan, Y., Wang, X., Li, J., Qin, R., & Wang, F. Y. (2018). An Overview of Smart Contract: Architecture, Applications, and Future Trends. *IEEE Intelligent Vehicles Symposium, Proceedings*, *2018-June*(Iv), 108–113. https://doi.org/10.1109/IVS.2018.8500488

Wang, Z., Yang, L., Wang, Q., Liu, D., Xu, Z., & Liu, S. (2019). ArtChain: Blockchain-enabled platform for art marketplace. *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, 447–454. https://doi.org/10.1109/Blockchain.2019.00068

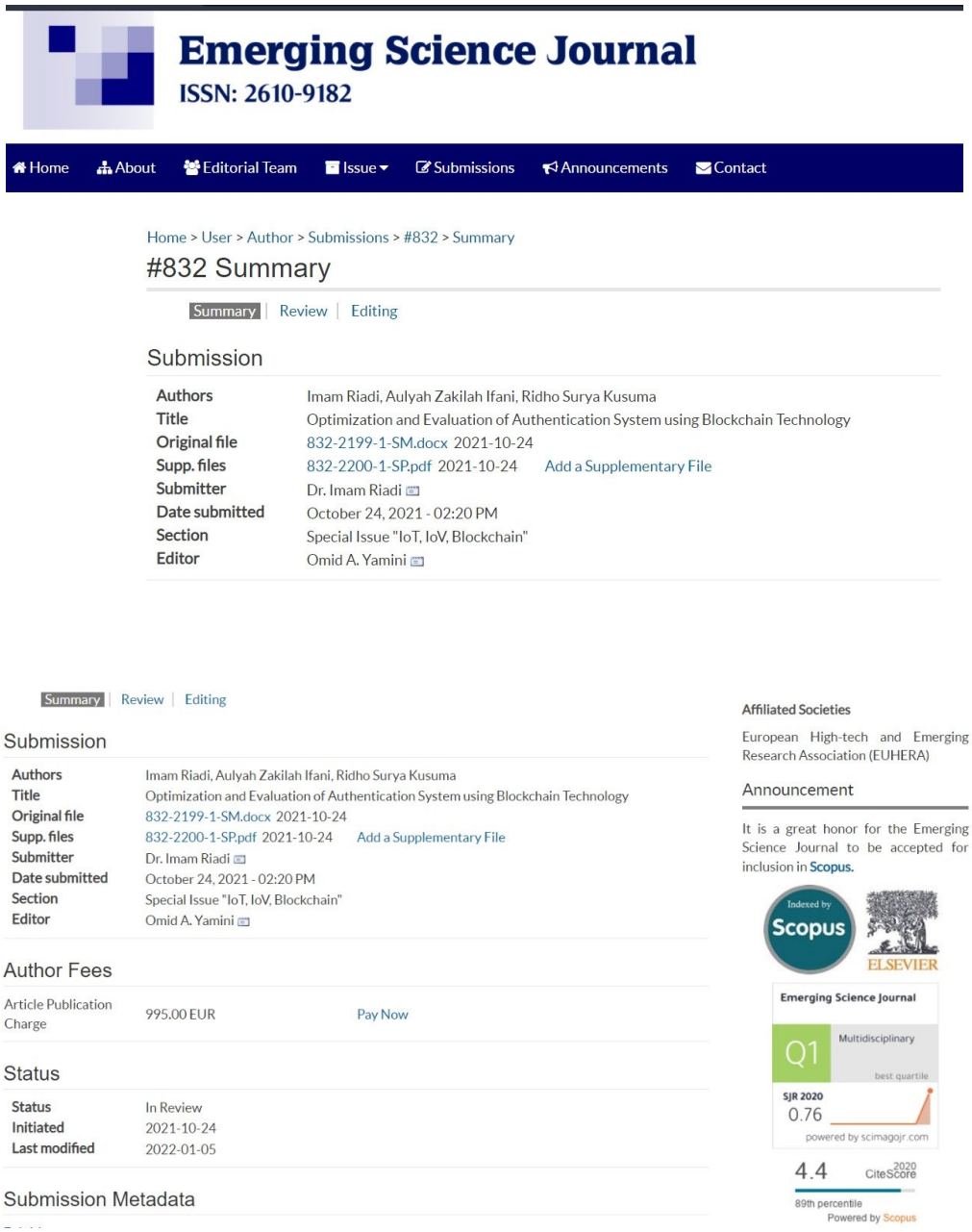
Yang, X., Chen, Y., & Chen, X. (2019). Effective scheme against 51% attack on proof-of-work blockchain with history weighted information. *Proceedings - 2019 2nd IEEE International Conference on Blockchain, Blockchain 2019*, 261–265. https://doi.org/10.1109/Blockchain.2019.00041

Zhang, R., Xue, R., & Liu, L. (2019). Security and privacy on blockchain. *ArXiv*, *1*(1).

Zheng, Y., Li, Y., Wang, Z., Deng, C., Luo, Y., Li, Y., & Ding, J. (2019). Blockchain-based privacy protection unified identity authentication. *Proceedings - 2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2019*, 42–49. https://doi.org/10.1109/CyberC.2019.00017

Available online at www.ijournalse.org
Emerging Science Journal
(ISSN: 2610-9182)
Vol. x, No. x, xxxxxxxxx, 20xx

c. Jurnal ESJ



Emerging Science Journal
ISSN: 2610-9182

Home | About | Editorial Team | Issue | Submissions | Announcements | Contact

Home > User > Author > Submissions > #832 > Summary

#832 Summary

Summary | Review | Editing

Submission

Authors	Imam Riadi, Aulyah Zakilah Ifani, Ridho Surya Kusuma
Title	Optimization and Evaluation of Authentication System using Blockchain Technology
Original file	832-2199-1-SM.docx 2021-10-24
Supp. files	832-2200-1-SP.pdf 2021-10-24 Add a Supplementary File
Submitter	Dr. Imam Riadi
Date submitted	October 24, 2021 - 02:20 PM
Section	Special Issue "IoT, IoV, Blockchain"
Editor	Omid A. Yamini

Summary | Review | Editing

Submission

Authors	Imam Riadi, Aulyah Zakilah Ifani, Ridho Surya Kusuma
Title	Optimization and Evaluation of Authentication System using Blockchain Technology
Original file	832-2199-1-SM.docx 2021-10-24
Supp. files	832-2200-1-SP.pdf 2021-10-24 Add a Supplementary File
Submitter	Dr. Imam Riadi
Date submitted	October 24, 2021 - 02:20 PM
Section	Special Issue "IoT, IoV, Blockchain"
Editor	Omid A. Yamini

Author Fees

Article Publication Charge	995.00 EUR	Pay Now

Status

Status	In Review
Initiated	2021-10-24
Last modified	2022-01-05

Submission Metadata

Affiliated Societies

European High-tech and Emerging Research Association (EUHERA)

Announcement

It is a great honor for the Emerging Science Journal to be accepted for inclusion in Scopus.

Indexed by Scopus

ELSEVIER

Emerging Science Journal

Q1 Multidisciplinary best quartile

SJR 2020 0.76

powered by scimagojr.com

4.4 CiteScore 2020

89th percentile

Powered by Scopus

S

Optimization and Evaluation of Authentication System using Blockchain Technology

Iman Riadi[1*], Aulyah Zakilah Ifani[2], Ridho Surya Kusuma[3]

[1] *Department of Information System, Universitas Ahmad Dahlan, Yogyakarta 55164, Indonesia*

[2,3] *Department of Informatic, Universitas Ahmad Dahlan, Yogyakarta 55164, Indonesia*

Abstract

User data security innovation is of particular concern in maintaining one's privacy rights, one of the severe violations when attackers can bypass user authentication so that it looks like something legitimate and legal. Based on this, this research seeks to optimize and evaluate blockchain-based authentication systems to minimize data leakage, manipulation, and modification. This research explores and evaluates authentication systems through a blockchain network's network forensic development life cycle (NFDLC) framework. This framework consists of five stages, namely initiation, acquisition, implementation, operations, and disposition. The results show that the percentage of authentication optimization reaches a value of 90.1%, and 8.9% is the percentage for evaluating systems such as the possibility of cyberattacks. Based on these results, this research has followed its objectives and can be helpful in further research.

Keywords: Authentication, Blockchain, NFDLC, Network, Cyberattacks;

Article History:

Received:

Revised:

Accepted:

Published:

1- Introduction

The development of the internet in this day and age is something that internet users need. The big challenge in using the internet is the presence of illegal attacks to gain access to a system [1]. There is much research that starts to discuss wireless, for example, by using MAC Address Filter [2][3], Service Set Identifier (SSID) [4], Extensible Authentication Protocol (EAP) [5], even captive portal [6].

Process modifying login authentication system using blockchain and innovative contract. The framework used is the Network Forensic Development Life Circle (NFDLC), while testing uses XSS, Burpsuite, SQL Injection and DOS. Not properly secured systems will be vulnerable to attack, on the network side or directly to the system. The study used multiple attacks in testing that approached computer network behaviour through network traffic logs to reconstruct early events with new engineering attacks[7].

The login authentication system needs to be considered a security. Authentication is proof of identity [8]. This system usually uses an authentication process in the form of usernames and passwords. Need a system that provides early warning when attacks, such as DOS, burp suite, SQL injection, and XSS, attacks the user's site or website. Cross-site-scripting (XSS) is a gap in the system that can cause others who enter by exploiting the system [9]. DOS or commonly called Denial of Service, is one of the web attacks. This type of attack is generally done by blocking network traffic through many attacks and can occur on many computers [10][11].

Blockchain-based authentication occurs when a user logs into the system. Authentication can not be used as a guideline to secure the system from DOS, Burpsuite, and XSS attacks, so this research uses blockchain technology and smart contracts in building the system. Hackers will find it difficult to corrupt data, especially by changing or modifying data [12]. That is because all computers have the same data. After all, blockchain technology has a decentralized nature which means it takes much time for hackers to crack the code on each block [13].

* CONTACT: imam.riadi@is.uad.ac.id

DOI: http://dx.doi.org/10.28991/esj-20XX-XXXXX

Some previous studies that use blockchain technology include research on blockchain applications with the integration of smart contracts in crowdfunding systems. The study explained that the system is centralized, not providing data on embezzlement activities and complete transparency. The fundraising users and funders have pointed out that the functional needs of the system can implement the following use case design [14]. Research on blockchain cryptocurrency technology (cryptocurrency) aims to illustrate the opportunities obtained from the technology that plays a role in cryptocurrencies [15]. Research on security and privacy on blockchain [16]. Research on school service top-up transaction data security using blockchain technology. The results obtained from this research are that modifying cryptography and blockchain technology can function correctly on the system [17]. Various applications of blockchain technology are about health, the Internet of Things (IoT), fundraising, digital asset management, education, and many others [18][19][20]. In improving e-commerce systems [16] *and* the e-voting system [21] using blockchain.

The login system requires authentication to secure the user's identity, so the study used blockchain technology to secure data from users and the framework used in supporting system development and addressing system weaknesses, namely the Network Forensic Development Life Circle (NFDLC). Blockchain can give confidence to third parties who oversee the process between sellers and buyers to confirm the authenticity of data and information [22]. The study aims to secure login authentication systems using blockchain technology from Burpsuite, XSS, SQL Injection, and DOS attacks based on previous research.

2- Research Method

A variety of applications can be using a blockchain techno-based login authentification system in intelligent payment applications, cryptocurrencies, healthcare systems, and any other using blockchain technology.

2-1- Blockchain Technology

The blockchain is a continuous strand of data blocks that list the previous data block from a new data. Each block records a set of related metadata data transactions. The blockchain network cannot tamper-proof and store these blocks of data and the data in every computer participating in it. Satoshi Nakamoto first implemented blockchain in 2008 as a peer-to-peer money exchange system. Nakamoto refers to transactional tokens exchanged between clients in his system as Bitcoin [23][24]. The first blockchain creates transactions between A and B records without the use of intermediaries. The cost of transactions with blockchain is much cheaper compared to traditional ways involving intermediaries. Transactions using blockchain are much more secure blocks [25].

Blockchain is not a stand-alone technology but is a configuration of many technologies, tools, and methods that address specific issues or use cases [26]. Blockchain technology separates itself from traditional centralized approaches that make it possible to securely manage chain data across a network of distributed and interconnected nodes [27]. The bitcoin cryptocurrency is the first application of blockchain underlying the transaction recording mechanism [28]. In bitcoin uses the concept of blockchain, which is a solution to the problem of the lack of third parties or intermediaries from financial institutions. Distributed bookkeeping technology, commonly called Distributed Ledger Technology (DLT), interprets the concept of blockchain wherein its implementation, every connected person in a network has a privilege to access its block [29]. The concept of distributed databases is that data is stored and distributed to each network when recorded data or information. The technology describes the method of eliminating third parties (financial institutions) in cryptocurrencies. The blockchain concept can also prevent double transactions because if someone changes a block in the blockchain or manipulates a block (The transaction duplicates in this case), the hash value will become invalid. After all, it does not store a valid value, i.e. the hash value of the previous block [30].

After validation and consensus decision, the previous block hash links each blockchain works [30] cryptographically. When the mining process creates a new block successfully, the data on the previous block will be difficult to change or manipulate. Storing data or transactions on the blockchain will be stored in the form of hashes. The hash form is hexadecimal except for storage, as a pointer that connects blocks using hashes that have a function. It can generate and validate new blocks [31]. Figure 1 is the format for each block.

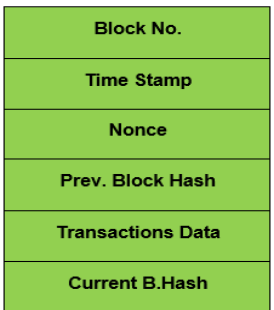


Figure 1: Format Block

Figure 1 shows the contents of each block. Block No, which represents block numbers, quickly identifies any block in the ledger. That timestamp is also an essential factor in finding any block of the entire ledger. Nonce to provide consensus among nodes as well as create identical hashes from different ledgers. Previous blocks connect one block to another and create a list of links from blocks in the ledger that make data irreversible. The amount of transaction data stored through the Markle tree concept quickly find any transaction from the block. The current block hash indicates the integrity of the block hash functions such as MD5, SHA 256. The Genesis Block becomes the first block of a ledger with no hash blocks and previous transactions [31].

2-2- Smart Contract

Invented by Nick Szabo in the mid-1990s, Nick Szabo recommends converting contract clauses into code and including them in software and hardware for automated execution, minimizing contract costs between the parties to transactions and preventing unwanted errors and malicious behaviour during the processing of contracts [32]. A smart contract is a record of self-executed computer transactions that facilitate and support the validation of any contract. Smart contracts have a code function consisting of a complete set of turning operations used to create contracts. After calling the contract, each contract will save the transaction into a decentralized database and cannot be changed [33]—an integrated contract procedure for databases of blockchain-operated for managing and transferring digital assets. Blockchain runs database programs for managing and transferring digital assets. The programming language for building smart contracts is the solidity programming language [34].

A characteristic of smart contracts is that a program or code runs on a blockchain platform, and a machine can read that code; The smart contract is part of a special program in the application. Once the Smart Contract is available. Distributing smart contracts [35][36]. Figure 2 is a smart contract mechanism.

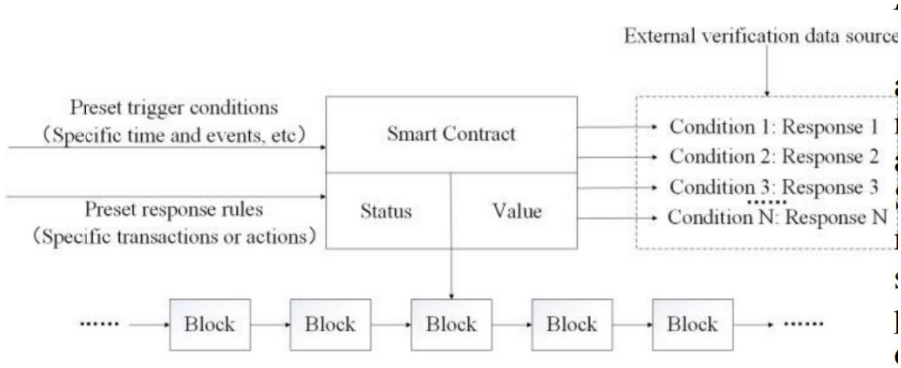


Figure 2: Smart Contract Mechanism [36]

Figure 2 shows the smart contract process mechanism where after all parties sign the smart contract. It is attached to the blockchain as a program code (such as a Bitcoin transaction), propagated over the P2P network, verified by nodes, and registered. Smart contracts consist of a predefined set of transition states and rules, situations that trigger the execution of a contract, such as the occurrence of a specific time or inevitable event, a response in a particular situation, and any others. Blockchain monitors the real-time status of smart contracts and executes contracts when certain activity conditions are met [36].

2-3- Burpsuite

Burpsuite is an open-source tool for performing or testing security on an application or system [37]. PortSwigger company creates burp suite using the java programming language. The ability to intercept HTTP becomes a priority on the burp suite [38][39]. The primary function of the burp suite is to intercept and display HTTP messages in a structured manner. The burp suite gives the tester a brief overview of the target system, all messages and parameters sent. In addition, the burp suite provides a GUI that provides complete control over all messages – drop, forward, repeat, modify, send later, and any others., so testers can design different attack scenarios and execute manually through the burp suite.

The results of the game can be viewed directly and analyzed directly by the tester [40]. Burpsuite has a commercial version, such as XSS, brute force, and others[41]. The study used a free version of the burp suite. This research does not require professional features.

2-4- Cross-Site Scripting (XSS)

Cross-Site Scripting (XSS) is an attack that attacks web applications. XSS can steal information from users [42]. The vulnerability's web application could allow cybercriminals to inject their malicious code into their TV displayed for end-users to receive. There are three XSS attacks, including XSS, Reflected-XSS, and DOM-based XSS attacks [43].

These attacks can be good on the client-side, server, or both [44]. Analyze process: There are three XSS-based defence approaches static, dynamic, and hybrid [45].

(a) Static analysis approach: untuk mengetahui kontrol data pada saat runtime sebelum menjalankan program.

(b) Dynamic analysis approach: it is done during the agreement at runtime after the software is released. This approach analyzes the data obtained during program execution.

(c) Hybrid approach: approaches that use both are static and dynamic.

2-5- Denial-of-Service (DOS)

Denial-of-Service is a type of attack on a computer or server on an Internet network that depletes a computer's resources until the computer can no longer function properly, indirectly preventing other users from accessing the computer services attacked. [46]. The most common DoS attacks aim to deplete network bandwidth, CPU cycles, or memory on the target system to unavailable services to legitimate users [47]. Another DOS goal to flood the resources of the target system is the first. Second, the exploration and exploitation of weak points in the system [48].

The types of DOS attacks most widely used by attackers are Syn-Spoofing, UDP-Flooding, HTTP-Flooding, Slowloris, Slow post, ICMP Echo, Brute Force.

(a) SYN-Spoofing: attacks target server tables that manage connections to clients.

(b) UDP-Flooding: This attack targets a specific port on the victim's system or server and then floods that port with a UDP packet to overload that port and stop the provided port.

(c) HTTP-Flooding: HTTP protocol attack, in which multiple bots flood a web server by assigning HTTP requests that deplete memory resources.

(d) Slowloris: an attack in which an attacker makes multiple connections to the server due to incomplete HTTP requests.

(e) Slow post: is an HTTP attack that sends an HTTP request to the victim's server.

(f) ICMP Echo: an attack that floods the server with an echo request.

(g) Brute Force: is attacks carried out on computer security systems by using passwords as authentication [49].

2-6- SQL Injection

SQL Injection is an attack technique against vulnerabilities or vulnerabilities owned by SQL [50]. A technique to discover vulnerabilities on websites that cause a hacker to influence SQL queries submitted through a database website [51]. SQLI attacks usually use a single quote character (') or double quote character (-) or fence sign (#) at the end of the number parameter to find out if the website is vulnerable or not [50].

2-7- Network Forensic Development Life Cycle (NFLDC)

Network Forensics Development Life Cycle (NFDLC) is a combination of the Information Systems Development Life Circle (ISDLC) and Network Forensic Readiness (NFR) methods. The concept of NFR maximizes the ability to gather credible evidence while minimizing the cost of inside response. The concept is a recommendation to increase the efficiency of the investigation. However, there is little to discuss how to integrate NFR into a network of systems. NFR appears to investigate malicious online intruders. In this case, NFR is a case study. They design ISDLC to incorporate security throughout the system development cycle. The ISDLC methods of each phase are analyzed and modified to include additional steps that create digital forensic embedding. Specific ISDLC modifications result in NFLDC [52][53]. The function of using this framework is to evaluate the authentication system in the use of blockchain technology. The use of NFDLC in research to evaluate the authentication system so that the results in network forensics serve as a reference in evaluating the authentication system and enabling the system's sustainability to be better.

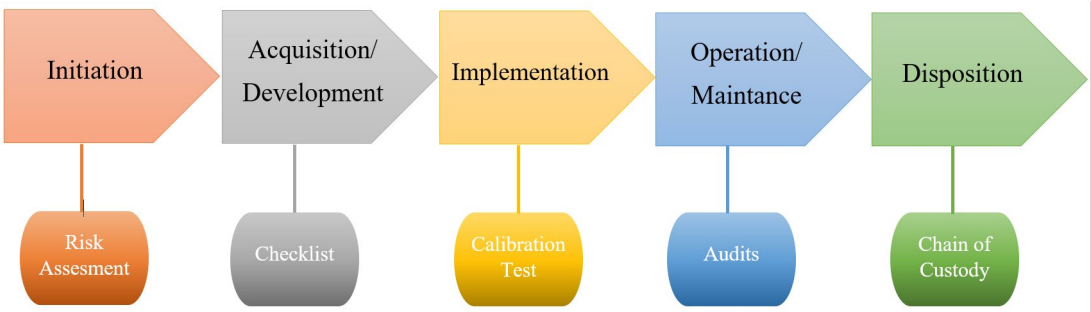


Figure 3: Framework NFLDC

Figure 3 shows the NFLDC framework conducting the stages of research. The stages consist of 5 parts, namely Initiation, Acquisition, Implementation, Operation, Disposition:

(a) Initiation: This stage is called the preparation stage. This stage is the process of developing all the needs that exist in this research. Preparation is needed for the investigation process to run smoothly, including determining the software to be used in the investigation process to obtain effective results and planning the actions to be taken during the investigation.

(b) Acquisition: The acquisition stage is the stage of designing system development, flowchart design. Design, in this case, is used to support the creation of the system. The system design stage allocates the needs of the hardware and software system to form the entire system architecture.

(c) Implementation: This stage is the stage of changing the design made on the system running on demand. This stage is to implement software design as a series of programs or program units. Testing involves verifying that each unit meets user specifications. Users perform tests to ensure that they are compatible with the software.

(d) Operation/Maintenance: This stage is the longest. The system has been installed and used. Maintenance includes correcting errors missed in the previous stage based on the evaluation results. The system evaluation in this study uses Burpsuite, XSS, SQL Injection, and DoS attack scenarios. Maintenance includes fixing errors that were not found in the previous stage based on the evaluation results to optimize the system better.

(e) Disposition: This stage is a stage to ensure the evidence obtained is appropriately stored. It Needs to be able to reuse it.

3- Result and Discussion

3-1- Initiation

Case simulations from this study use the Visual Studio Code tool. The result of the simulation stage is a system that is tested and run. The login system utilizes the Ethereum blockchain platform, which implements blockchain technology and smart contracts. As an application programming interface, Web3js connects the browser with an extension called a metamask, which acts as a bridge between the login system and the Ethereum blockchain. This metamask acts as an Ethereum wallet for information management. Meanwhile, to build smart contracts using the solidity programming language.

In this research stage, the user must be a member of the blockchain network; when he becomes a member, the user will have an ethereum account. An ethereum blockchain-based application must run a smart contract. So for smart contracts will be signed first by people who already have an ethereum account to run an Ethereum-based application. If the user wants to log in must be registered as a smart contract signer. Login data from users will be stored as a hash to the blockchain via smart contract. The User requests access to the website; for example, a hash derived from the user is given credentials compared to the hash stored in the Smart Contract. Therefore, when the user logins and the user data match, the user is authorized to access the web. Conversely, if it does not fit, then access is denied. Each user must connect with an Ethereum address previously done in the registration process, as this address generates the user's login hash.

3-2- Acquisition

The flowchart is the first thing in doing this research because it is the research workflow to be built. Figure 4 is a system flowchart when performing operations.

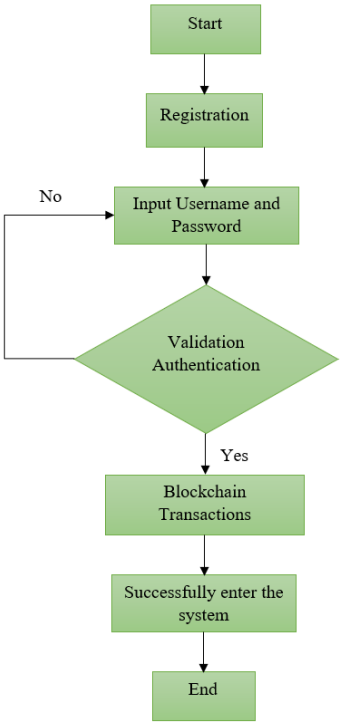


Figure 4: Flowchart System

Figure 4 shows the flowchart of the blockchain system, where the user registration first and then inputs the username and password. The data entered is then authenticated validation. The purpose is to find out the authenticity of data from users. The system will verify; after all, the user will perform blockchain transactions. After that, it made it into the system. Figure 5 shows the registration flow.

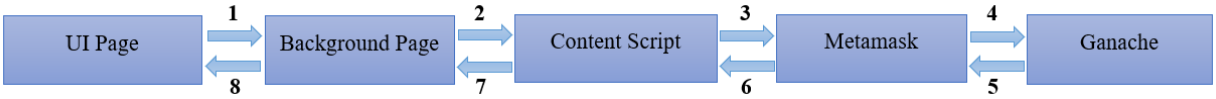


Figure 5: System Registration Flow

Figure 5 is the system registration flow. The workflow of registration for users does not yet have an account. Deploy commands will be sent to the user, then the contract is forwarded to the browser. The metamask browser will detect the contract. Metamask will disseminate the contract after confirming, then the identification of the contract is reversed when used. Identification appears on the website after the identifier is detected, then returns to the metamask extension. Metamask stores contract identifiers, and now data can be used. When the user already has an account, the user will log in. Gambat 6 shows the flow of the login.

Figure 6: Login Procedure Flow

Figure 6 shows the flow of the login procedure; after the registration process is complete, the user will go to the login page in chrome. Chrome storage identifies the contract from the metamask extension storage. Furthermore, the background page contains contract, timestamp and timestamp flags to the content script. Figure 7 shows the security scenario of the login system.

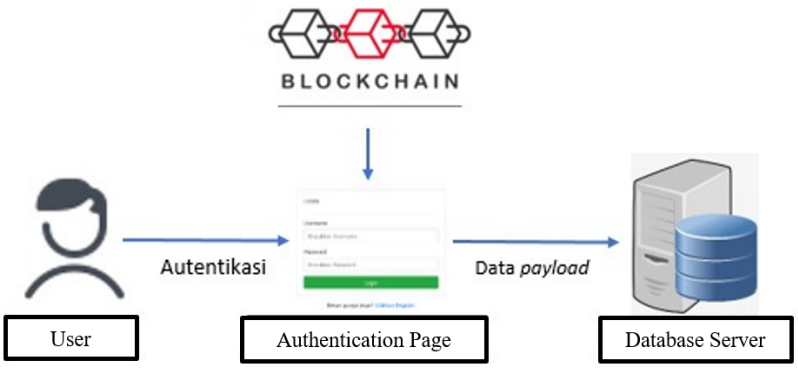


Figure 7: Login System Security Scenario

Figure 7 is a security scenario where the user performs the process of filling in the username and password, and then the system performs the authentication process. If the user wants to log in must be registered as a smart contract signer. The user will store the login data as a hash derived from the user's credentials, then compare it with the hash in the smart contract. When the user logs into the system and the data is appropriate, then the user is authorized to access the web; otherwise, access is denied, such as for testing the login system against cyber-attacks using several attack strategies against the Ethereum blockchain system consisting of several scenarios, namely Burpsuite, XSS, SQL Injection, and DDoS as shown in Figure 8.

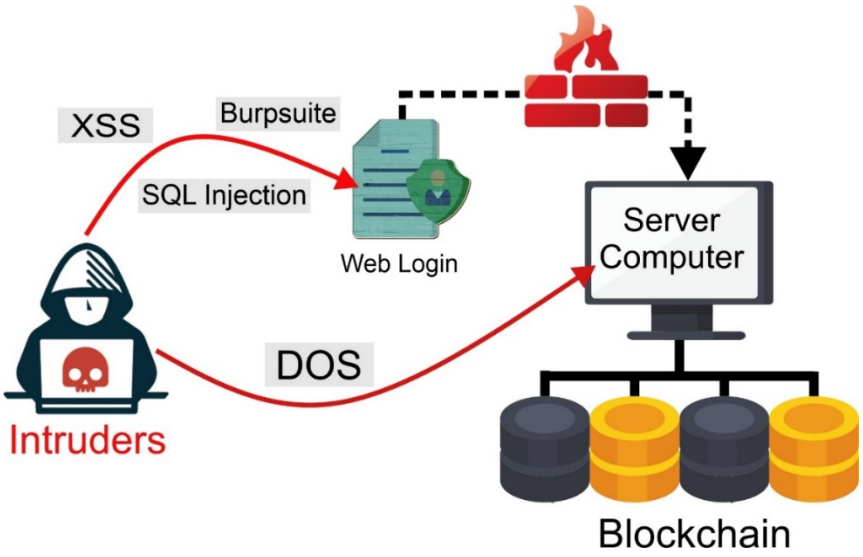


Figure 8: Scenario of Cyber Attack Against Login Website

Figure 8 provides an overview of the flow of cyberattacks on testing this system. The purpose of burp suite, XSS, and SQL injection attacks is to obtain any security breaches that can occur, attacks using Denial of Service (DOS) to test the capabilities of the server infrastructure against indiscriminate attacks, thus degrading system performance. Attack strategies in this study as in the formula (1).

$$D^*_p \in \arg\max_{D_p \in \varphi(D_p)} \mathcal{A}(D_p, \boldsymbol{\theta}) \qquad (1)$$

Formula (1), variable D_p∈φ (D_p), is a set of cyber threats used by an attacker. Variable A(D_p, θ) is a function to represent the attacker's objectives [40]. The system test seeks to find system optimization problems by considering each vulnerability and several cyber threat models.

3-3- Implementation

This section describes the implementation of blockchain where this research uses the Ethereum blockchain platform that implements blockchain technology and smart contracts. As an application programming interface, web3js connects the browser with an extension called metamask, which acts as a bridge between the login system and the ethereum blockchain. This metamask acts as an ethereum wallet for information management. Meanwhile, the following smart contract code uses the solidity programming language in Table 1.

Table 1. Building a Smart Contract

Program Code

```
contract Authentication {
uint256 public nbOfUsers;
struct user {
 string signature hash;
  address user address;
contract Authentication {
}
mapping(address => User) private user;
    constructor() {
   nbOfUsers = 0;
}
    function register(string memory _signature) public {
 require(
     user[msg.sender].userAddress ==
         address(0x0000000000000000000000000000000000000000),
      "already registered"
  );
user[msg.sender].signatureHash = _signature;
  user[msg.sender].userAddress = msg.sender;
  nbOfUsers++;
}
    function getSignatureHash() public view returns (string memory) {
return user[msg.sender].signatureHash;}
 function getUserAddress() public view returns (address) {
  return user[msg.sender].userAddress; }}
```

Table 1 is the source code for creating smart contracts. After executing the contract on the blockchain, the contract can provide services to the use. The contract transaction will be directed to the contract-related address when the user registers, as in the registration function. In this case, the MSG variable is a place to store all data information. The contract then processes the recipient's data with the process written in the registration function. At the time of registration, the user must be a member of the blockchain network. First, the user creates an account before accessing the network. Ganache becomes the place to get networked. In this study, users must become members of the blockchain network first create an account before accessing the network. Ganache becomes the place to get networked. Ethereum blockchain-based applications must run smart contracts. Smart contracts will be signed first by people who already have an Ether ID to run Ethereum-based applications. Ganache already provides 10 default accounts, and each account has a balance of 100 eth. This account is for transactions on the ganache blockchain. Figure 9 shows the ethereum address in the Ganache.

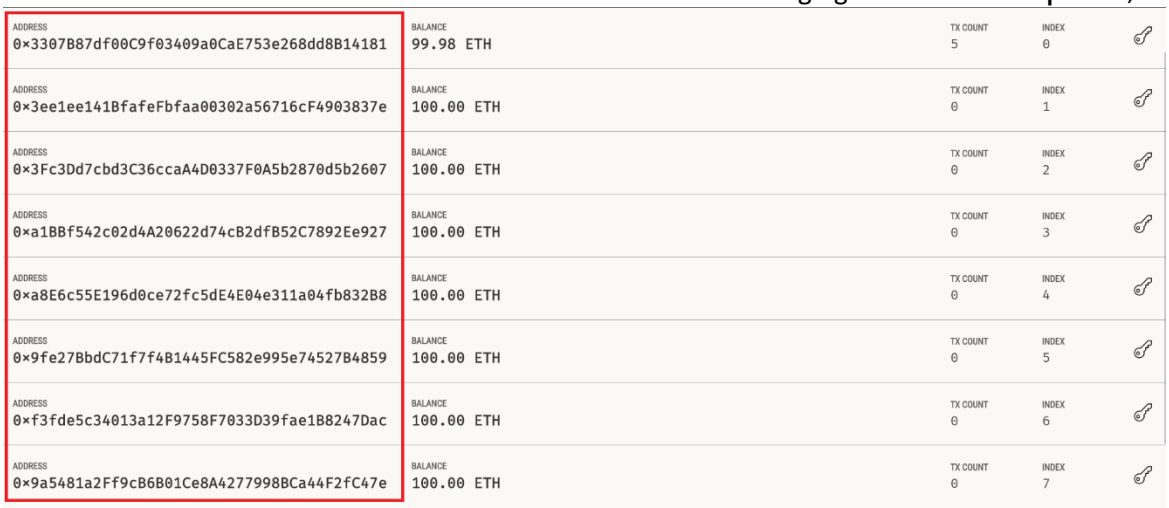


Figure 9: Address Ethereum Ganache

Ganache already provides 10 default accounts, and each account has a balance of 100 eth (Figure 9). This account is for transactions on the ganache blockchain one network for one registration process. So if the user wants to register a new account to enter the login system, then the user takes one ethereum network on the Ganache. If the user wants to log in, the user must have registered as a signer of the smart contract. Before the User logs into the system page, the user first connects the Ganache to metamask. This metamask is a bridge for users to log into the web browser. This metamask allows users to run ethereum Dapps directly in the browser.

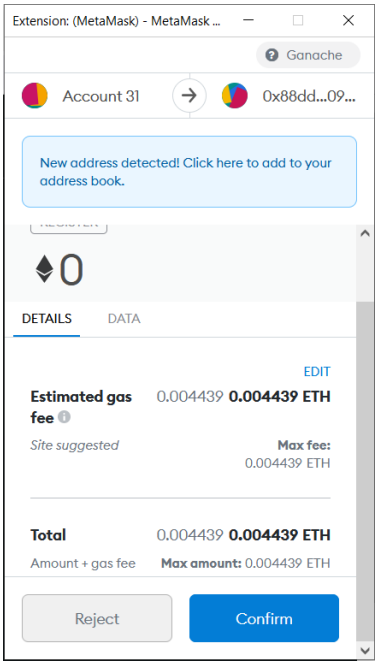


Figure 10: Blockchain Transaction Process

Figure 10 provides information on the transaction process through a legitimate currency storage application. This research uses Ganache which acts as a database to store records of all transactions on the Ethereum network. The following is a login screen via the truffle program, as shown in Figure 11.

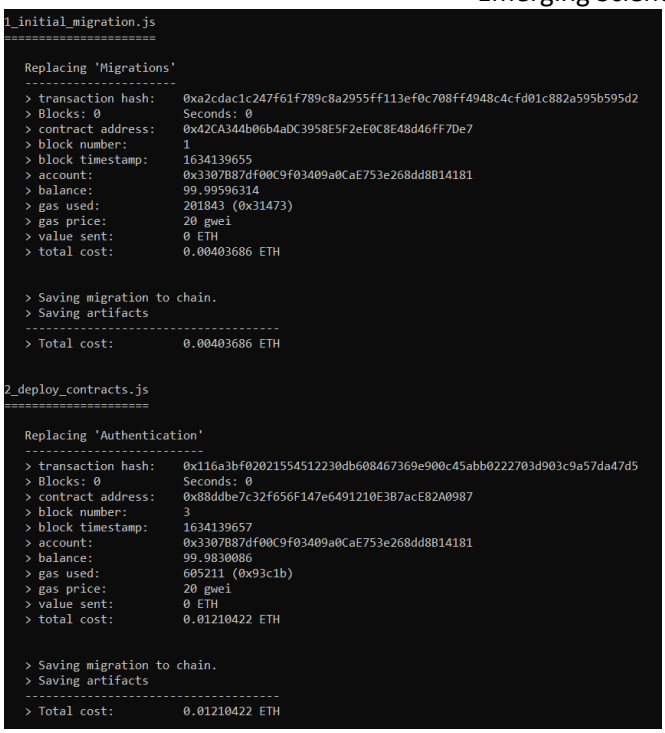


Figure 11: Contract Deployment Results

The migration script goes into the migration folder. Truffles execute scripts in those folders using lexicographic sequences. This folder, by default, contains 1_initiation_migration.js scripts, which migrate Migration.sol contracts, useful for Truffles (Figure 11). Blockchain will store blogs that have registered and carry out transactions until the process is complete. Figure 12 shows a record of successful transactions on the blockchain.

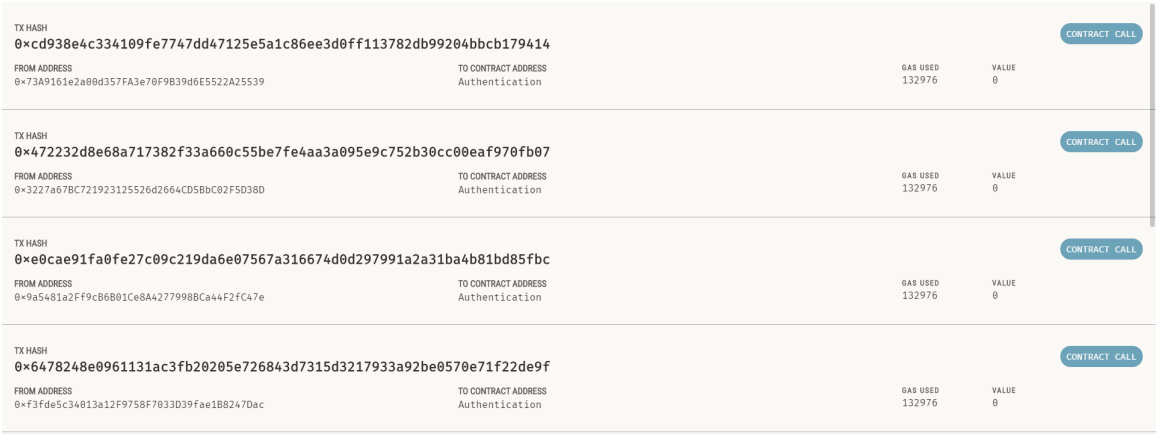


Figure 12: Successful Transactions Recorded on the Blockchain

Note that on the Blockchain (Figure 12), users can see the gas used. After the transaction is successful, users can log in to the system using the correct username, password. After that, the user managed to get into the system. Table 2 is the record of the resulting tx hash.

Table 2. Hash Transaction

Block	Tx Hash	Gas Used
1	TX 0x3028abd60f6b9dffbe944f1358109e074437fbb6626c7bbb2bd6b444c552f606	605211
2	TX 0x37e11c23c9eadfeb60c9624d89b160f2b55456f735bce9a8eb307729991abada	14796

3	TX 0xfd4ff9b826779977439ec0b58f53e280040378d2e96178184bf4278e2a70869b	132976
4	TX 0x983fc20847ca06847630824a63cb7ccd4da0ae7c8859c49442ce9e27fe0993b8	132976
5	TX 0xc336ccfe700354cc5ba74a7b30f957b21e3ae0264bc6359fc12f28abc342f1d2	132976
6	TX 0x83c97365d2214f08d050477e019052e1f92e8cf75c2152738f88561214161a84	132976
7	TX 0x6478248e0961131ac3fb20205e726843d7315d3217933a92be0570e71f22de9f	132976
8	TX 0xe0cae91fa0fe27c09c219da6e07567a316674d0d297991a2a31ba4b81bd85fbc	132976
9	TX 0x472232d8e68a717382f33a660c55be7fe4aa3a095e9c752b30cc00eaf970fb07	132976
10	TX 0xcd938e4c334109fe7747dd47125e5a1c86ee3d0ff113782db99204bbcb179414 132976	132976

Table 2 shows the tx hash and gas used generated after entering the username and password on the system. The registration process generates a hash value on the Ganache. Ganache is a place to store interconnected blocks.

3-4- Operation

This section describes the test results using Burpsuite, XSS, SQL Injection, and DDOS. The login page in this study only uses blockchain without any security code. Here are the results of the attack scenario.

- Burpsuite attack scenario in testing this system using version 2021.8.2 community edition has not made a significant impact.
- Scenario SQL Injection attack attempts to violate security by entering into the database system on the website. This attack attempt only gives an "Incorrect Login" response repeatedly.
- The XSS attack scenario in this study used the injection code <script src="https://10.10.10.8:3000/hook.js"></script>, inject code into any URL or website elements. The injection code attack on the URL address was not successful, so the website page only provided a refresh response display, following the results of the XSS injection attack in the login form section looks as in Figure 13.

Figure 13: XSS Injection Attack on Login Website

Figure 13 provides information on the results of the XSS attack. The attack managed to make the website page into an error due to an undefined AccountAddress. Based on these results, applying secure code in the login form section is necessary to propagate or verify that the input value entered is appropriate.

- The DOS attack scenario in this study seeks to degrade the performance of blockchain servers. The attack process uses FLOOD SYN by executing the command "hping3 -c 15000 -d 120 -S -w 64 -p 80 --flood --rand-source 10.10.10.6", DOS attacks run for 30 seconds with a total network traffic log of 2,372,850 log lines. Here is a graph of the DOS attack on the webserver blockchain as in Figure 14.

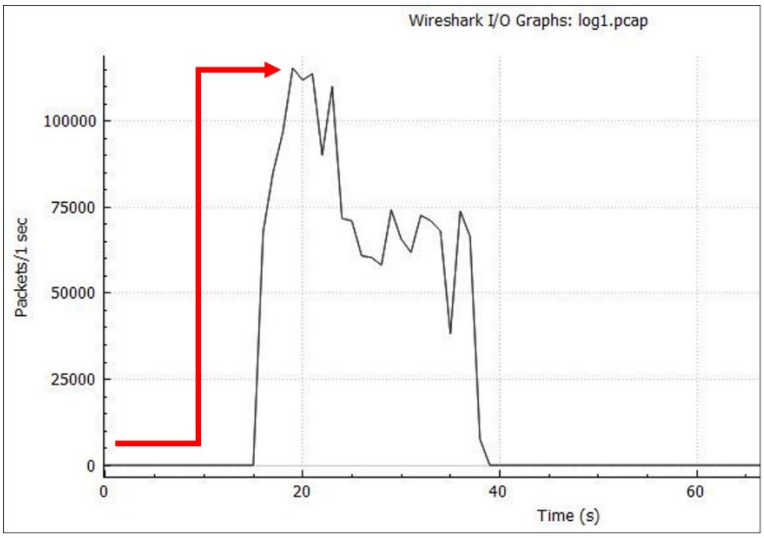


Figure 14: DOS Attack on Blockchain Web Server

Figure 14. It shows a pattern of network behaviour that is surging due to DOS cyberattacks. Here are the details of the DOS attack as in Figure 15.

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s
Frame	100.0	2372859	100.0	412870946	35 M	0	0	0
Ethernet	100.0	2372859	8.0	33220026	2883 k	0	0	0
Internet Protocol Version 4	100.0	2372811	11.5	47456220	4119 k	0	0	0
User Datagram Protocol	0.0	1	0.0	8	0	0	0	0
NetBIOS Datagram Service	0.0	1	0.0	216	18	0	0	0
SMB (Server Message Block Protocol)	0.0	1	0.0	134	11	0	0	0
SMB MailSlot Protocol	0.0	1	0.0	25	2	0	0	0
Microsoft Windows Browser Protocol	0.0	1	0.0	48	4	1	48	4
Transmission Control Protocol	100.0	2372806	80.5	332192840	28 M	2372806	332192840	28 M
Internet Control Message Protocol	0.0	4	0.0	256	22	4	256	22
Address Resolution Protocol	0.0	48	0.0	1380	119	48	1380	119

Figure 15: Details of DOS Attack on Blockchain Web Server

Figure 15 provides detailed information from the NETWORK BEHAVIOR DOS attack on the web server, in the red box indicating that the network protocol used is the Transmission Control Protocol (TCP). DOS attacks result in decreased web server performance due to full system memory, so the server must restart to operate again.

3-5- Disposition

Blockchain technology and smart contracts successfully build a login authentication system. User login data is stored as a hash to the blockchain through a smart contract. Smart contracts will be signed first by people who already have an Ether ID to run Ethereum-based applications. Hackers cannot change and manipulate the data contained in each block because every participating or connected computer stores all that data.

Testing this system uses Wireshark tools. One relevant form of implementation applied to support documentation is to use the external metadata approach .pcap to store Wireshark files to secure data for users. This disposition stage shows that the application of blockchain technology to the login authentication system can secure user data. User data becomes accurate because it encrypts and converts it into hash form.

Testing for attack scenarios using multiple scenarios shows that the burp suite's attack scenario has not worked. Burpsuite has not displayed significant data. Unlike SQL Injection, XSS and DoS attack scenarios have a significant impact.

4- Discussion

After implementing blockchain technology, the data becomes secure as it turns into a hash form to calculate the results of monitoring the number of logs on the blockchain system using formula 1.

$$E = 100 - \left(\frac{U_{BC} * NoS}{\sqrt{ReDB}}\right) \quad (1)$$

Where U_BC is the number of blockchain users, NoS is the average number of blockchain logs. ReDB is the average number of overall logs [54]. The results obtained can be seen in the Table. 3

Table 3. Log List

Block	Log Line (NoS)	Overall Log Row (ReDB)
1	24	3734
2	52	2181
3	24	870
4	26	895
5	22	2153
6	22	658
7	20	715
8	25	731
9	20	709
10	22	741

Table 3 presents log results after using blockchain. NoS has an average number of log lines of 51.4 logs, while ReDB has an average number of overall logs of 2,677 logs. Therefore, the percentage of results obtained as much as 90.1%.

5- Conclusion

Blockchain technology and smart contracts successfully built a login authentication system. The login system uses the PHP programming language to build the system interface, while the blockchain implementation uses a solid programming language. Login authentication systems using blockchain can secure data, this is evidenced by testing Wireshark based on the resulting data logs, with a 90.1% percentage of test results showing a relatively high system security level. Tests that implement multiple attack scenarios result in the burp suite attack scenario not being successfully used. Burpsuite has not displayed significant data. Unlike SQL Injection, XSS and DoS attack scenarios have a significant impact.

6- Declaration

6-1- Author Contributions

Imam Riadi (I.R.), Aulyah Zakilah Ifani (A.Z.I) and Ridho Surya Kusuma (R.S.K.), Conceptualization, I.R., A.Z.I and R.S.K.; methodology, I.R.; software, A.Z.I.; validation, I.R., A.Z.I and R.S.K.; formal analysis, I.R.; investigation, R.S.K.; resources, I.R.; data curation, A.Z.I., R.S.K.; writing—original draft preparation, I.R.; writing—review and editing, A.Z.I., R.S.K.; visualization, I.R.; supervision, I.R.; project administration, A.Z.I., R.S.K.; funding acquisition, I.R. All authors have read and agreed to the published version of the manuscript.

6-2- Funding

Universitas Ahmad Dahlan funds this Research.

6-3- Acknowledgements

The authors would like to express appreciation and gratitude to Universitas Ahmad Dahlan for funding this research.

6-4- Conflict and Interest

The author declares that there is no conflict of interest in connection with the publication of this manuscript. In addition, the authors have thoroughly examined ethical issues, including plagiarism, informed consent, errors, fabrication or falsification of data, double publication or submission, and redundancy.

7- References

[1] M. Rusdan and M. Sabar, “Analisis dan Perancangan Jaringan Wireless Dengan Wireless Distribution System Menggunakan User Authentication Berbasis Multi-Factor Authentication,” *Jt. (Journal Inf. Technol.*, vol. 02, no. 01, pp. 17–24, 2020.

[2] Z. M. Subekti and S. Subandri, “Implementasi Metode Per Connection Queue Dengan Access User Direct Mac Filtering Pada Jaringan Wireless,” *INOVTEK Polbeng - Seri Inform.*, vol. 5, no. 2, p. 240, 2020, doi: 10.35314/isi.v5i2.1472.

[3] Y. Tian, N. Zheng, X. Chen, and L. Gao, "Wasserstein Metric-Based Location Spoofing Attack Detection in WiFi Positioning Systems," *Secur. Commun. Networks*, vol. 2021, 2021, DOI: 10.1155/2021/8817569.

[4] F. Teferi and J. S. Nixon, "A Security Mechanism to Mitigate DDoS Attack on Wireless Local Area Network (WLAN) using MAC with SSID," *Int. J. Comput. Sci. Eng.*, vol. 7, no. 4, pp. 864–869, 2019, doi: 10.26438/ijcse/v7i4.864869.

[5] T. N. Hidayat and I. Riadi, "Optimation Wireless Security IEEE 802.1X using the Extensible Authentication Protocol-Protected Extensible Authentication Protocol (EAP-PEAP)," *Int. J. Comput. Appl.*, vol. 174, no. 11, pp. 25–30, 2021, DOI: 10.5120/ijca2021920988.

[6] N. Marques, A. Zúquete, and J. P. Barraca, "Integration of the Captive Portal paradigm with the 802.1X architecture," pp. 1–28, 2019.

[7] R. Umar, I. Riadi, and R. S. Kusuma, "Mitigating Sodinokibi Ransomware Attack on Cloud Network Using Software-Defined Networking (SDN)," *Int. J. Saf. Secur. Eng.*, vol. 11, no. 3, pp. 239–246, Jun. 2021, DOI: 10.18280/ijsse.110304.

[8] U. Rahardja, E. P. Harahap, and D. D. Christianto, "Pengaruh Teknologi Blockchain Terhadap Tingkat Keaslian Ijazah," *Technomedia J.*, vol. 4, no. 2, pp. 211–222, 2019, doi: 10.33050/tmj.v4i2.1107.

[9] Y. Cui, J. Cui, and J. Hu, "A Survey on XSS Attack Detection and Prevention in Web Applications," *ACM Int. Conf. Proceeding Ser.*, pp. 443–449, 2020, DOI: 10.1145/3383972.3384027.

[10] S. T. Vimala and J. P. M. Dhas, "SDN based DDoS attack detection system by exploiting ensemble classification for cloud computing," *Int. J. Intell. Eng. Syst.*, vol. 11, no. 6, pp. 282–291, 2018, DOI: 10.22266/IJIES2018.1231.28.

[11] H. F. El-Sofany, "A new cybersecurity approach for protecting cloud services against DDoS attacks," *Int. J. Intell. Eng. Syst.*, vol. 13, no. 2, pp. 205–215, 2020, doi: 10.22266/ijies2020.0430.20.

[12] Z. Wang, L. Yang, Q. Wang, D. Liu, Z. Xu, and S. Liu, "ArtChain: Blockchain-enabled platform for art marketplace," *Proc. - 2019 2nd IEEE Int. Conf. Blockchain, Blockchain 2019*, pp. 447–454, 2019, DOI: 10.1109/Blockchain.2019.00068.

[13] T. Salman, M. Zolanvari, A. Erbad, R. Jain, and M. Samaka, "Security services using blockchains: A state of the art survey," *IEEE Commun. Surv. Tutorials*, vol. 21, no. 1, pp. 858–880, 2019, DOI: 10.1109/COMST.2018.2863956.

[14] F. Aprialim, Adnan, and A. W. Paundu, "Penerapan Blockchain dengan Integrasi Smart Contract pada Sistem Crowdfunding," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 1, pp. 148–154, 2021, doi: 10.29207/resti.v5i1.2613.

[15] R. C. Noorsanti, H. Yulianton, and K. Hadiono, "Blockchain - Teknologi Mata Uang Kripto (Crypto Currency)," *Pros. SENDI_U*, vol. 3, no. November, p. 306, 2018.

[16] R. Zhang, R. Xue, and L. Liu, "Security and privacy on blockchain," *arXiv*, vol. 1, no. 1, 2019.

[17] A. Fadlil, I. Riadi, and A. Nugrahantoro, "Data Security for School Service Top-Up Transactions Based on AES Combination Blockchain Technology," *Lontar Komput. J. Ilm. Teknol. Inf.*, vol. 11, no. 3, p. 155, 2020, doi: 10.24843/lkjiti.2020.v11.i03.p04.

[18] L. Ismanto, H. S. Ar, A. N. Fajar, Sfenrianto, and S. Bachtiar, "Blockchain as E-Commerce Platform in Indonesia," *J. Phys. Conf. Ser.*, vol. 1179, no. 1, 2019, DOI: 10.1088/1742-6596/1179/1/012114.

[19] A. Rizky, S. Kurniawan, R. D. Gumelar, V. Kurniawan, and M. B. Prakoso, "Use Of blockchain technology in implementing information system security on education," *BEST (Journal Biol. Educ. Sains Technol.*, vol. 4, no. 1, pp. 62–70, 2021.

[20] Y. Zheng *et al.*, "Blockchain-based privacy protection unified identity authentication," *Proc. - 2019 Int. Conf. Cyber-Enabled Distrib. Comput. Knowl. Discov. CyberC 2019*, pp. 42–49, 2019, DOI: 10.1109/CyberC.2019.00017.

[21] A. Alam, S. M. Zia Ur Rashid, M. Abdus Salam, and A. Islam, "Towards Blockchain-Based E-voting System," *2018 Int. Conf. Innov. Sci. Eng. Technol. ICISET 2018*, pp. 351–354, 2018, DOI: 10.1109/ICISET.2018.8745613.

[22] S. Shorman, M. Allaymoun, and O. Hamid, "Developing the E-Commerce Model a Consumer To Consumer Using Blockchain Network Technique," *Int. J. Manag. Inf. Technol.*, vol. 11, no. 02, pp. 55–64, 2019, DOI: 10.5121/ijmit.2019.11204.

[23] S. Gupta and M. Sadoghi, "Encyclopedia of Big Data Technologies," *Blockchain Trans. Process.*, no. May 2020, DOI: 10.1007/978-3-319-63962-8.

[24] D. Efanov and P. Roschin, "The all-pervasiveness of the blockchain technology," *Procedia Comput. Sci.*, vol. 123, pp. 116–121, 2018, DOI: 10.1016/j.procs.2018.01.019.

[25] P. Abas Sunarya, Henderi, Sulistiawati, A. Khoirunisa, and P. Nursaputri, "Blockchain family deed certificate for privacy and data security," *2020 5th Int. Conf. Informatics Comput. ICIC 2020*, no. January 2021, 2020, DOI: 10.1109/ICIC50835.2020.9288528.

[26] A. Rejeb, E. Sűle, and J. G. Keogh, "Exploring new technologies in procurement," *Transp. Logist. Int. J.*, vol. 18, no. 45, pp. 76–86, 2018.

[27] M. Milkovic, J. Samardžija, and M. Ognjan, "Application of Blockchain Technology in Media Ecology," *Medijska Istraz.*, vol. 26, no. 1, pp. 29–52, 2020, DOI: 10.22572/mi.26.1.2.

[28] F. Sartipi, "Publicizing construction firms by cryptocurrency," *J. Constr. Mater.*, vol. 2, no. 3, pp. 1–8, 2021, DOI: 10.36756/jcm.v2.3.1.

[29] A. Rejeb and K. Rejeb, "LogForum," vol. 16, no. 3, pp. 363–372, 2020.

[30] C. Reviews, "The Future of the Digital Economy and SAP's Role," vol. 7, no. 8, pp. 1812–1818, 2020.

[31] M. Arse and J. Dubey, "A Survey of Internet of Things node's transactions Secure through Blockchain Technology," *Int. J. Comput. Appl.*, vol. 175, no. 25, pp. 33–37, 2020, DOI: 10.5120/ijca2020920796.

[32] D. Lo and X. D. Le, "Institutional Knowledge at Singapore Management University Smart contract development : Challenges and opportunities," 2019.

[33] N. Chaniago, P. Sukarno, and A. A. Wardana, "Electronic document authenticity verification of diploma and transcript using a smart contract on the ethereum blockchain," *Regist. J. Ilm. Teknol. Sist. Inf.*, vol. 7, no. 2, pp. 149–163, 2021, DOI: 10.26594/REGISTER.V7I2.1959.

[34] S. Bragagnolo, H. Rocha, M. Denker, and S. Ducasse, "SmartInspect: Solidity smart contract inspector," *2018 IEEE 1st Int. Work. Blockchain Oriented Softw. Eng. IWBOSE 2018 - Proc.*, vol. 2018-Janua, pp. 9–18, 2018, DOI: 10.1109/IWBOSE.2018.8327566.

[35] B. K. Mohanta and D. Jena, "An Overview of Smart Contract and Use Cases in Blockchain Technology: 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)," *2018 9th Int. Conf. Comput. Commun. Netw. Technol.*, pp. 1–4, 2018.

[36] S. Wang, Y. Yuan, X. Wang, J. Li, R. Qin, and F. Y. Wang, "An Overview of Smart Contract: Architecture, Applications, and Future Trends," *IEEE Intell. Veh. Symp. Proc.*, vol. 2018-June, no. Iv, pp. 108–113, 2018, DOI: 10.1109/IVS.2018.8500488.

[37] K. V. V. N. L. S. Kiran, R. N. K. Devisetty, N. P. Kalyan, and K. Mukundini, "ScienceDirect ScienceDirect ScienceDirect Building an Intrusion Detection System for IoT Environment using Machine Learning Techniques Building an Intrusion Detection System for IoT Environment using Machine Learning Techniques," *Procedia Comput. Sci.*, vol. 171, no. 2019, pp. 2372–2379, 2020, DOI: 10.1016/j.procs.2020.04.257.

[38] C. Pallavi, R. Girija, and S. L. Jayalakshmi, "An Analysis on Network Security Tools and Systems," *SSRN Electron. J.*, 2021, DOI: 10.2139/ssrn.3833455.

[39] G. S. T and D. Sasikala, "Vulnerability Assessment of Web Applications using Penetration Testing," no. 4, pp. 1552–1556, 2019, DOI: 10.35940/ijrte.B2133.118419.

[40] L. F. Sikos, *AI in Cybersecurity*. 2018.

[41] K. Patel, "A survey on vulnerability assessment penetration testing for secure communication," *Proc. Int. Conf. Trends Electron. Informatics, ICOEI 2019*, no. Icoei, pp. 320–325, 2019, DOI: 10.1109/ICOEI.2019.8862767.

[42] F. M. M. Mokbal, W. Dan, A. Imran, L. Jiuchuan, F. Akhtar, and W. Xiaoxi, "MLPXSS: An Integrated XSS-Based Attack Detection Scheme in Web Applications Using Multilayer Perceptron Technique," *IEEE Access*, vol. 7, pp. 100567–100580, 2019, DOI: 10.1109/ACCESS.2019.2927417.

[43] S. Gupta and B. B. Gupta, "Cross-Site Scripting (XSS) attacks and defence mechanisms: classification and state-of-the-art," *Int. J. Syst. Assur. Eng. Manag.*, vol. 8, pp. 512–530, 2017, DOI: 10.1007/s13198-015-0376-0.

[44] Y. Khera, D. Kumar, S. Sujay, and N. Garg, "Analysis and Impact of Vulnerability Assessment and Penetration Testing," *Proc. Int. Conf. Mach. Learn. Big Data, Cloud Parallel Comput. Trends, Perspectives Prospect. Com. 2019*, pp. 525–530, 2019, DOI: 10.1109/COMITCon.2019.8862224.

[45] U. Sarmah, D. K. Bhattacharyya, and J. K. Kalita, "A survey of detection methods for XSS attacks," *J. Netw. Comput. Appl.*, vol. 118, pp. 113–143, 2018, DOI: 10.1016/j.jnca.2018.06.004.

[46] M. M. Shurman, R. M. Khrais, and A. A. Yateem, "IoT denial-of-service attack detection and prevention using hybrid IDS," *Proc. - 2019 Int. Arab Conf. Inf. Technol. ACIT 2019*, pp. 252–254, 2019, DOI: 10.1109/ACIT47987.2019.8991097.

[47] H. F. El-Sofany, S. A. El-Seoud, and I. A. T. F. Taj-Eddin, "A case study of the impact of denial of service attacks in cloud applications," *J. Commun.*, vol. 14, no. 2, pp. 153–158, 2019, DOI: 10.12720/jcm.14.2.153-158.

[48] N. F. Syed, Z. Baig, A. Ibrahim, and C. Valli, "Denial of service attack detection through machine learning for the IoT," *J. Inf. Telecommun.*, vol. 4, no. 4, pp. 482–503, 2020, DOI: 10.1080/24751839.2020.1767484.

[49] M. Abushwereb, M. Mustafa, M. Al-kasassbeh, and M. Qasaimeh, "Attack based DoS attack detection using multiple classifiers," no. Mid, 2020.

[50] D. Chen, Q. Yan, C. Wu, and J. Zhao, "SQL Injection Attack Detection and Prevention Techniques Using Deep Learning," *J. Phys. Conf. Ser.*, vol. 1757, no. 1, 2021, DOI: 10.1088/1742-6596/1757/1/012055.

[51] F. Q. Kareem *et al.*, "SQL Injection Attacks Prevention System Technology: Review," *Asian J. Res. Comput. Sci.*, no. July, pp. 13–32, 2021, DOI: 10.9734/ajrcos/2021/v10i330242.

[52] B. E. Endicott-Popovsky and D. A. Frincke, "Embedding forensic capabilities into networks: Addressing inefficiencies in digital forensics investigations," *Proc. 2006 IEEE Work. Inf. Assur.*, vol. 2006, pp. 133–139, 2006, DOI: 10.1109/iaw.2006.1652087.

[53] B. Endicott-Popovsky, D. A. Frincke, and C. A. Taylor, "A theoretical framework for organizational network forensic readiness," *J. Comput.*, vol. 2, no. 3, pp. 1–11, 2007, DOI: 10.4304/jcp.2.3.1-11.