ANTIVIRUS: Jurnal Ilmiah Teknik Informatika (p – ISSN: 1978 – 5232; e – ISSN: 2527 – 337X)
Vol. 15 No. 2 November 2021, pp. 242 – 251

PERANCANGAN *SINGLE SIGN ON* (SSO) PADA APLIKASI WEB MENGGUNAKAN *CLOUD IDENTITY* (STUDI KASUS: POLITEKNIK NEGERI TANAH LAUT)

Fathurrahmani[1)], Herpendi[2)], Khairul Anwar Hafizd[3)]
[1, 2, 3)]Teknik Informatika, Politeknik Negeri Tanah Laut
e-mail: fathurrrahmani@politala.ac.id[1)], herpendi@politala.ac.id[2)], hafizd@politala.ac.id[3)]

Abstrak **:** *Politeknik Negeri Tanah Laut memiliki beberapa sistem berbasis web yang telah diimplementasikan. Sistem-sistem yang dibangun tersebut masih bersifat standalone dan belum terintegrasi, sehingga pengguna harus memiliki akun yang berbeda pada masing-masing sistem. Pengguna harus mengingat setiap akun untuk mengakses sistem dan untuk alasan keamanan biasanya pengguna mengganti passwordnya secara rutin. Proses pergantian pada Password ini akan memerlukan waktu yang cukup lama mengingat setiap perubahan yang dilakukan berbanding lurus dengan jumlah sistem yang ada (existing). Oleh karena itu diperlukan sistem yang bisa mengintegrasikan akun pengguna dan mengelola proses otentikasi dan otorisasi. Proses ini membutuhkan sebuah unit server untuk tambahan yang menjadi media penghubung antara sistem layanan aplikasi dengan sistem integrator. Tujuan dari penelitian ini adalah menerapkan sebuah inovasi sistem yang bisa menangani seluruh otentikasi dan otorisasi setiap sistem aplikasi dan dikenal dengan sistem Single Sign On (SSO). Sehingga bisa ditarik manfaat penelitian dari adanya sistem Single Sign On pengguna hanya cukup dengan menggunakan satu akun pengguna bisa mengakses banyak sistem tanpa memasukkan Username dan Password berulang. Penerapannya data akun pengguna diambil dari Cloud Identity melalui Secure LDAP, kemudian data pengguna dikelola oleh RADIUS Server dan didistribusikan ke sistem layanan aplikasi yang ada (existing). Penelitian telah berhasil dilakukan dan diimplementasikan pada website yang dimiliki oleh Politeknik Negeri Tanah Laut, dengan diterapkannya Single Sign On maka login ke website hanya dengan menggunakan Username dan Password yang sama.*

Kata Kunci— Otentikasi, Cloud Identity, LDAP, Otorisasi, RADIUS, SSO

Abstract **:** *Politeknik Negeri Tanah Laut has several web-based systems that have been implemented. The systems built are still standalone and not yet integrated, so users must have different accounts on each system. Users must remember each account to access the system and for security reasons users usually change their passwords regularly. This Password change process will take a long time considering that every change made is directly proportional to the number of existing systems. Therefore we need a system that can integrate user accounts and manage the authentication and authorization process. This process requires an additional server that acts as a liaison between the system integrator and the application service system. The purpos of this research is to create an innovative system that can handle all authentication and authorization of each application system and is known as the Single Sign On (SSO) system. So that the benefits of research from the existence of a Single Sign On system, users only by using one user account can access many systems without entering repeated Usernames and passwords. In practice, user account data is retrieved from Cloud Identity via Secure LDAP, then user data is managed by the RADIUS Server and distributed to existing application service systems. The research has been successfully carried out and implemented on a website owned by the Politeknik Negeri Tanah Laut, with the implementation of Single Sign On, login to the website only by using the same Username and password.*

Keywords— Authentication, Cloud Identity, LDAP, RADIUS, SSO

I. PENDAHULUAN

DUNIA teknologi berkembang dengan sangat cepat dan pesat. Teknologi banyak dikembangkan untuk berbagai macam keperluan bagi penggunanya. Teknologi mempunyai manfaat yang sangat besar salah satunya dalam pengolahan data dan informasi. Pemanfaatan itu akan membantu sebuah pekerjaan seperti halnya pengolahan data yang lebih cepat, keputusan yang akan diambil menjadi lebih tepat, hingga menghemat waktu dan biaya. Teknologi saat ini dibutuhkan di berbagai instansi, perusahaan, komunitas dan perguruan tinggi salah satunya adalah Politeknik Negeri Tanah Laut (Politala)

DOI: https://doi.org/10.35457/antivirus.v15i2.1813

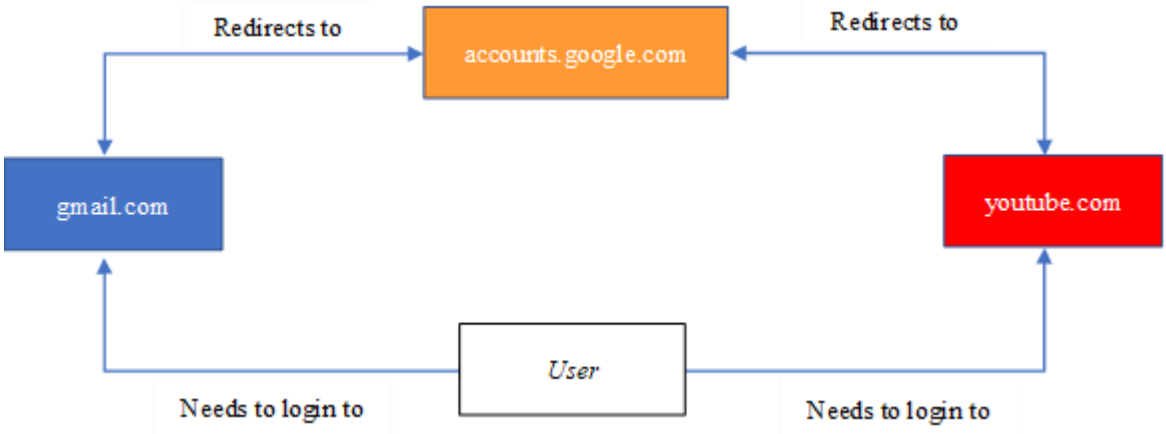
khususnya dalam bidang teknologi.

Politala telah mengembangkan beberapa layanan aplikasi berbasis web. Dari semua layanan yang dibangun setiap pengguna harus memiliki akun disetiap layanan sehingga pengguna harus mengingat setiap akun untuk mengakses sistem dan untuk alasan keamanan biasanya pengguna mengganti *password*-nya secara rutin. Proses pada pergantian *Password* ini tentunya akan memerlukan waktu cukup lama sebab mengingat setiap perubahan yang akan dilakukan berbanding lurus pada jumlah sistem yang telah ada (*existing*).

Berdasarkaan hal tersebut dibutuhkan sebuah sistem yang berfungsi mengintegrasikan keseluruhan layanan pada aplikasi dan bisa mengelola proses otentikasi dan otorisasi masing-masing sistem layanan. Otentikasi ialah merupakan sebuah proses verifikasi yang bertujuan menentukan apakah seseorang pengguna memiliki hak untuk mengakses sistem aplikasi pada web atau tidak [1]. Pengguna yang tanpa melalui otentikasi disebut *anonymous* atau *guest*.

Cara paling sederhana yang bisa dilakukan ialah dengan menggunakan otentikasi login. Dimana seorang *user* atau pengguna menginputkan *Username* dan *Password* (*credential*), lalu selanjutnya akan dilakukan verifikasi oleh sistem. Verifikasi akan menetukan apakah *credential* tadi valid ataukah tidak valid, jika valid maka *user* tersebut boleh melakukan akses ke dalam sistem. Sebaliknya jika tidak valid maka *user* tidak memiliki hak untuk mengakses apapun ke dalam sistem [2].

Proses otentikasi pada sebuah sistem terintegrasi memerlukan sebuah perangkat tambahan yang berguna menjadi media penghubung antara sistem layanan aplikasi dengan sistem integrator. Sistem ini mampu menangani seluruh proses otentikasi pada setiap sistem aplikasi, sistem ini dikenal sebagai Sistem *Single Sign On* (SSO). Sistem ini ialah sebuah teknologi yang bisa mengizinkan *user* jaringan agar bisa melakukan akses sumber daya dalam jaringan tertentu hanya dengan menggunakan sebuah akun pengguna saja. Keuntungan dari sistem SSO ini ialah *user* tidak perlu banyak memiliki *Username* dan *Password* serta bisa memudahkan dalam pemrosesan data [3]. Sistem *Single Sign On* atau yang lebih dikenal dengan SSO ialah sebuah platform identitas serta manajemen data *user* yang memberikan pengelolaan, dan kemudahan serta kemanan pengguna. SSO memungkinkan *user* untuk masuk cukup sekali saja dan memiliki hak untuk melakukan akses ke semua aplikasi yang telah miliki [1]. Gambaran dari SSO bisa dilihat pada Gambar 1.



Gambar 1. Single Sign On

Sistem SSO memerlukan sebuah media penyimpanan yang dapat diakses secara cepat dalam operasinya. Media penyimpanan tersebut yang digunakan dalam penelitian ini ialah *Lightweight Directory Access Protocol* (LDAP). LDAP merupakan *protocol* yang mengatur bagaimana data *directory* bisa diakses secara cepat melalui suatu jaringan. Agar data pengguna dapat dilakukan otentikasi dan otorisasi maka memerlukan sebuah server yang mana dalam penelitian ini menggunakan server *Remote Authentication Dial-In User Service* (RADIUS). RADIUS adalah sebuah protokol keamanan komputer yang digunakan untuk melakukan otentikasi, otorisasi, dan pendaftaran akun pengguna secara terpusat untuk mengakses jaringan. RADIUS diterapkan dalam jaringan dengan model *client-server*. RADIUS ini bertugas menangani *Authentication, Authorization dan Accounting* (AAA).

Penelitian dalam hal pengembangan sistem otentikasi dan otorisasi menggunakan LDAP dan RADIUS sudah pernah dilakukan oleh beberapa peneliti diantara: Yuliansyah [2] mengimplementasikan sistem

DOI: https://doi.org/10.35457/antivirus.v15i2.1813

otentikasi dan otorisasi untuk proses login pada multi aplikasi web yang berbasis Bahasa pemrograman PHP dengan langkah mengoptimalkan penggunaan data dari RADIUS server. Hasil dari penelitian ini ialah pengguna pada beberpa aplikasi web berbasis PHP bisa dilakukan integrasi pengelolaannya dengan membangun sistem otentikasi dan otorisasi dengan RADIUS server melalui aplikasi FreeRADIUS. Proses optimalisasi RADIUS server sebagai sistem otentikasi dan otorisasi ini bisa membuat *user* hanya akan memiliki satu akun untuk beberapa aplikasi yang berbeda-beda.

Muttaqin [3] telah melakukan implementasi sistem otentikasi hotspot menggunakan LDAP dan RADIUS pada jaringan internet kampus di Teknik Sistem Komputer Univeristas Diponegoro. Setiap server bisa saling terintegrasi dan terhubung dengan jaringan internet kampus, Server RADIUS bisa melakukan akses ke *database* LDAP server menggunakan media Radtest. Proses pada otentikasi *hotspot* menggunakan antarmuka *login captive portal covachilli* yang berfungsi memblok jaringan lokal sehingga client tidak diberikan izin untuk masuk pada jaringan internet di kampus sebelum login.

Qidri [4] juga melakukan implementasi sistem otentikasi dan otorisasi untuk proses login ketika pengguna akan menggunakan internet. Implementasi sistem ini dibuat dalam aplikasi web berbasis PHP dengan cara mengoptimalkan penggunaan dari freeRADIUS. Pengguna dikelola menggunakan aplikasi berbasis web dengan menggunakan bahasa pemrograman PHP dan MySQL sebagai basis datanya.

Dari ketiga penelitian diatas bisa dilakukan inovasi penerapan SSO untuk memaksimalkan pengembangan sistem otentikasi dan otorisasi dan disesuaikan dengan kebutuhan di Politala. Seluruh civitas akademika Politala memiliki akun email institusi yang berafiliasi dengan Google Suite Edu dimana otomatis memiliki Cloud Identity. Hal ini bisa dimanfaatkan sebagai data akun pengguna yang nantinya akan digunakan untuk *login* ke beberapa layanan aplikasi berbasis web termasuk layanan internet dikampus melalui suatu *captive portal*. Secure LDAP yang tersedia di Google Suite akan dijadikan penghubung ke server RADIUS. RADIUS akan menjadi integrator yang mengelola proses otentikasi dan otorisasi layanan aplikasi berbasis web dan layanan internet. RADIUS juga akan dintegrasikan dengan Mikrotik sebagai pengelola lalu lintas jaringan LAN dan internet di Politala.

II. TINJAUAN PUSTAKA

A. *Single Sign On* (SSO)

Single Sign On merupakan suatu mekanisme yang membuat *user* hanya perlu mengingat sebuah *Username* dan *Password* yang otentik untuk membuka beberapa layanan atau sistem sekaligus. Teknologi SSO adalah sebuah teknologi yang mengizinkan *user* jaringan agar bisa mengakses sumber daya pada jaringan hanya dengan menggunakan cukup satu akun pengguna saja [5]. Teknologi ini sangat menarik dan diminati, khususnya dalam pengelolaan jaringan yang sangat besar dan bersifat heterogen di saat sistem operasi dan aplikasi yang digunakan oleh computer. Pengguna ini berasal dari macam-macam vendor, dan pengguna diminta untuk mengisi informasi tentang dirinya ke dalam tiap platform yang berbeda tersebut yang mana hendak diakses oleh pengguna.

Sistem SSO adalah sebuah sistem terpusat yang berbasis pada server, tidak ada aksess lain selain SSO untuk bisa mengakses sebuah jaringan terstruktur yang terintegrasi, SSO merupakan sebuah sistem proteksi dan pengamanan data tingkat tinggi yang bisa memberikan kenyamanan dan kepercayaan lebih terhadap sebuah sistem jaringan internet.

Salah satu contoh dari sistem SSO ialah protokol Kerberos, yang telah diimplementasikan ke dalam sistem operasi Windows 2000 ke atas. Protokol yang sama bisa juga digunakan di dalam kumpulan sistem operasi UNIX. Novell juga menawarkan fungsi SSO miliknya sendiri yang sering disebut sebagai Novell Single Sign On (NSSO) yang bisa digunakan dalam lingkungan Windows/NetWare. Beberapa perusahaan seperti Entrust Technologies dan RSA Security juga menawarkan fungsi SSO yang berbasis kriptografi sebagai kunci publik.

Dengan menggunakan teknologi SSO, seorang pengguna hanya cukup melakukan proses otentikasi satu kali saja untuk memperoleh izin akses terhadap semua layanan yang ada di dalam jaringan. Sistem SSO menghindari *login* yang ganda dengan cara melakukan identifikasi subjek secara ketat dan hanya memperkenankan informasi otentikasi untuk digunakan dalam sistem atau pada kelompok sistem yang terpercaya. Sistem SSO ini dapat meningkatkan kegunaan jaringan yang secara keseluruhan dan pada saat yang sama bisa melakukan pemusatan pengelolaan dari parameter sistem yang relevan. Pengguna layanan

DOI: https://doi.org/10.35457/antivirus.v15i2.1813

lebih meminati sistem SSO karena pengelola layanan jaringan memiliki banyak tugas tambahan yang harus dilakukan. Namun, penambahan tugas ini perlu perhatian ekstra untuk menjamin bukti-bukti otentikasi agar tidak tersebar dan tidak dilakukan penyadapan pihak lain ketika ada kegiatan melintasi jaringan.

Selain mendatangkan manfaat, dari cara pandang seperti ini beberapa orang pengamat memperkirakan bahwa penggunaan SSO bisa menghemat biaya untuk memelihara *password* yang rumit yang bisa mencapai ratusan dolar setiap pengguna tiap tahunnya. Tetapi, kenyataannya implementasi SSO dalam sebuah jaringan yang heterogen adalah rumit.

B. *Cloud Identity*

Cloud Identity merupakan produk Identitas sebagai Layanan (IDaaS) dan juga pengelolaan mobilitas perusahaan (EMM). Produk ini memberikan layanan identitas serta administrasi *endpoint* yang terpisah di Google Workspace sebagai bentuk produk mandiri. Sebagai administrator, pengguna bisa menggunakan Cloud Identity untuk mengelola data pengguna, aplikasi, dan perangkat dari satu lokasi secara terpusat, yaitu konsol Google Admin.

Cloud Identity Free Edition meliputi layanan pengelolaan endpoint dan identitas intinya. Cloud Identity Free Edition memberikan layanan Akun Google terkelola kepada pengguna yang tidak memerlukan layanan Google Workspace tertentu saja seperti Gmail, Google Sites dan Google Kalender. Namun begitu pengguna bisa mengakses Dokumen, Google Drive, Spreadsheet, Slide, Keep, dan Meet. Pengguna bisa menggunakan akun Cloud Identity beserta dengan layanan Google lainnya, seperti Chrome, Android Enterprise, Google Cloud dan berbagai aplikasi pihak ketiga lainnya.

C. Lightwieght Directory *Access* Protocol *(LDAP)*

Lightweight Directory Access protocol (LDAP) ialah Protokol yang digunakan untuk keperluan mengakses berbagai informasi dalam sebuah direktori. LDAP dikembangkan berdasarkan X.500 hanya saja lebih mudah serta mendukung TCP/IP. Walaupun penggunaannya secara luas tapi LDAP yang merupakan Open Protocol sangat dinamis karena bisa diimplementasikan pada aplikasi seperti Public Key dan *E-mail* dengan berbagai platform serta sistem operasi [6]. LDAP juga sebagai sebuah protokol yang mengatur mekanisme dalam mengkases layanan direktori (*Directory Service*) yang bisa digunakan untuk mendeskripsikan banyak informasi layaknya informasi tentang orang-orang, organisasi, aturan, layanan dan banyak entitas lainnya.

LDAP menggunakan model client-server yang mana client mencoba mencarnya pada *Directory Information Tree* (DIT) yang tersimpan pada server [7]. LDAP sering biasa digunakan di sistem *cloud*. LDAP bisa digunakan sebagai sumber otentikasi aplikasi jaringan seperti otentikasi mail *VPN server, file server, server* dan layanan *server* lainnya yang tentunya mendukung LDAP. Agar dapat membuat *server* LDAP bisa menggunakan *software* gratis seperti OpenLDAP (di Linux) atau dari Microsoft yaitu menggunakan Active Directory yang ada di Windows Server Comunity.

LDAP ini sebenarnya merupakan bagian dari Internet Protocol. LDAP digunakan untuk dapat mengakses suatu *directory* seperti *directory E-mail* suatu perusahaan, *directory* telepon dan lainnya. LDAP ini tidak hanya membaca informasi, tetapi juga bisa menambah dan memperbarui informasi yang ada directory tersebut. LDAP juga telah dilengkapi Simple Authentication and Security Layer (SASL) untuk memeriksa dan memastikan apakah suatu *user* memiliki hak dan diperbolehkan masuk atau tidak ke dalam sistem. Sebab itulah LDAP juga banyak digunakan untuk SSO, yaitu hanya dengan sekali *sign-on*, pengguna bisa mengakses berbagai aplikasi yang telah disediakan sistem. Sehubungan dengan penggunaan LDAP, Microsoft mengembangkan *Active Directory* (AD) yaitu suatu layanan yang mengendalikan komputer-komputer yang telah tergabung dalam Windows Domain. Komputer-komputer akan dapat login ke dalam *Active Directory*. AD akan memberikan serangkaian layanan *security policy*, *update software*, dan menginstal program ke dalam komputer-komputer *client*.

D. *RADIUS*

RADIUS atau dikenal dengan *Remote Authentication Dial-In User Service* ialah suatu protokol yang memungkinkan keamanan jaringan nirkabel untuk melakukan otentikasi, otorisasi, dan akuntansi untuk melakukan *remote user* yang ingin melakukan akses pada suatu sistem atau layanan dari pusat server

DOI: https://doi.org/10.35457/antivirus.v15i2.1813

jaringan komputer. RADIUS menjalankan sistem administrasi secara terpusat, sistem ini akan mempermudah tugas operator/administrator dalam mengelola *user* nirkabel LAN. Melalui sistem ini *user* sebagai pengguna jaringan *hotspot* bisa menggunakan *hotspot* pada tempat yang berbeda-beda hanya dengan melakukan otentikasi ke sebuah server RADIUS yang ada tanpa harus meminta *account* berulang-ulang kepada operator. Sarena satu *user* hanya menggunakan satu *Username* dan satu *Password* untuk semua jaringan *hotspot* yang telah menggunakan sistem otentikasi terpusat [8].

Free RADIUS adalah RADIUS server yang bersifat Open Source. FreeRADIUS mendukung semua protokol otentikasi dan dilengkapi dengan web administrasi pengguna berbasis PHP. FreeRADIUS awalnya dikembangkan oleh Alan Dekok dan Miquel Smoorenburg pada bulan Agustus 1999. Sebelum mengembangkan FreeRADIUS, Miquel sempat mengembangkan Cistron RADIUS server, namun tidak dikembangkan lagi hingga kini. Seiring perkembangan zaman FreeRADIUS terus dikembangkan dan didukung dengan banyak fitur selain dukungan teks *file* juga mendukung LDAP, MySQL, PostgreSQL, Oracle dan banyak fitur lainnya [9]. RADIUS sekarang telah dikembangkan untuk dapat melakukan otentikasi terhadap akses jaringan secara jarak jauh dengan cara menggunakan koneksi selain *dial-up*, seperti access point nirkabel, *switch ethernet*, *Virtual Private Networking* (VPN), dan perangkat lainnya.

E. *Otentikasi*

Otentikasi (*authentication*) ialah proses verifikasi data atau identitas seseorang atau pengguna perangkat tertentu saat hendak masuk ke dalam sebuah sistem. Proses verifikasi tersebut pada umumnya menggunakan *Username* dan *Password* pengguna yang nantinya akan dijadikan informasi untuk mengetahui apakah orang yang telah mengakses itu ialah pemilik akun asli yang secara sah dan berhak masuk ke dalam sistem. Otentikasi menetapkan data identitas satu pihak ke pihak lainnya. Kebanyakan pada umumnya otentikasi dilakukan dengan menetapkan identitas pengguna ke beberapa bagian dari sistem yang ada, biasanya melalui sarana kata sandi [10].

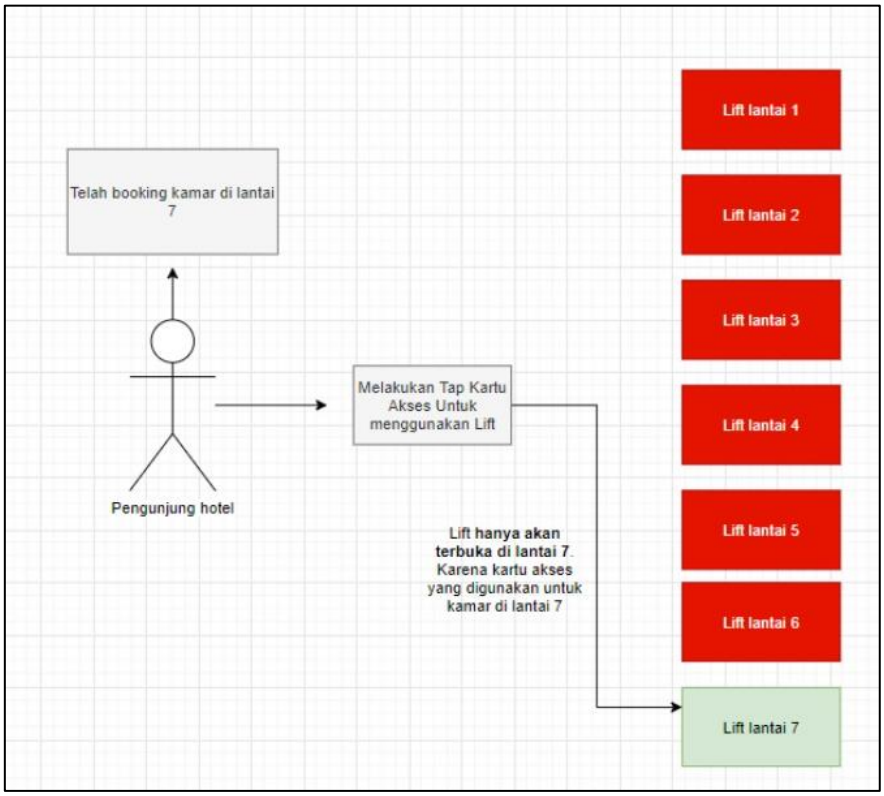
Otentikasi ialah salah satu dari banyak metode yang digunakan dalam rangka menyediakan bukti bahwa dokumen tertentu yang telah diterima secara elektronik benar datang dari orang yang bersangkutan serta tak berubah. Langkah kerjanya yaitu dengan mengirimkan sebuah kode tertentu melaui email, selanjutnya pemilik email memberikan balasan terhadap email tersebut atau mengetikan kode tertentu yang telah dikirimkan melalui email. Otentikasi server berfungsi dalam rangka mengenali pengguna yang berintegrasi ke jaringan dan memuat semua informasi dari pengguna tersebut, dalam praktek biasanya otentikasi server mempunyai back-up yang berguna untuk menjaga jika server itu terdapat masalah sehingga jaringan dan pelayanan tidak mengalami gangguan.

Dalam sebuah aplikasi Web dibutuhkan mekanisme yang bisa melindungi data dari pengguna yang tidak memiliki hak mengaksesnya, misalnya sebuah situs Web yang mana berisikan foto-foto keluarga serta hanya bisa diakses sesama anggota keluarga saja. Mekanisme ini bisa diimplementasikan dalam bentuk sebuah proses login yang umumnya terdiri dari tiga macam tahapan yaitu: identifikasi, otentikasi dan otorisasi.

F. *Otorisasi*

Otorisasi ialah proses penentuan hak akses layanan apa saja yang dapat kita terima setelah identitas kita diverifikasi. Berbeda dengan otentikasi yang memeriksa identitas pengguna. Proses otorisasi ini akan terjadi setelah *user* diidentifikasi atau telah terbukti dalam proses otentikasi *single factor* dan *two factor*. Proses otorisasi akan memberikan hak kepada seorang pengguna untuk bisa mengakses sumber daya pada aplikasi Contohnya, melihat data tertentu, mengakses menu tertentu, mengakses database tertentu, melakukan perubahan pada suatu direktori dan lain sebagainnya. Untuk menggambarkan proses otentikasi dan otorisasi adalah dengan menganalogikannya dengan sistem lift pada sebuah hotel. Contohnya ialah seorang pengujung memesan kamar hotel di lantai 7. Perhatikan Gambar 2.

DOI: https://doi.org/10.35457/antivirus.v15i2.1813



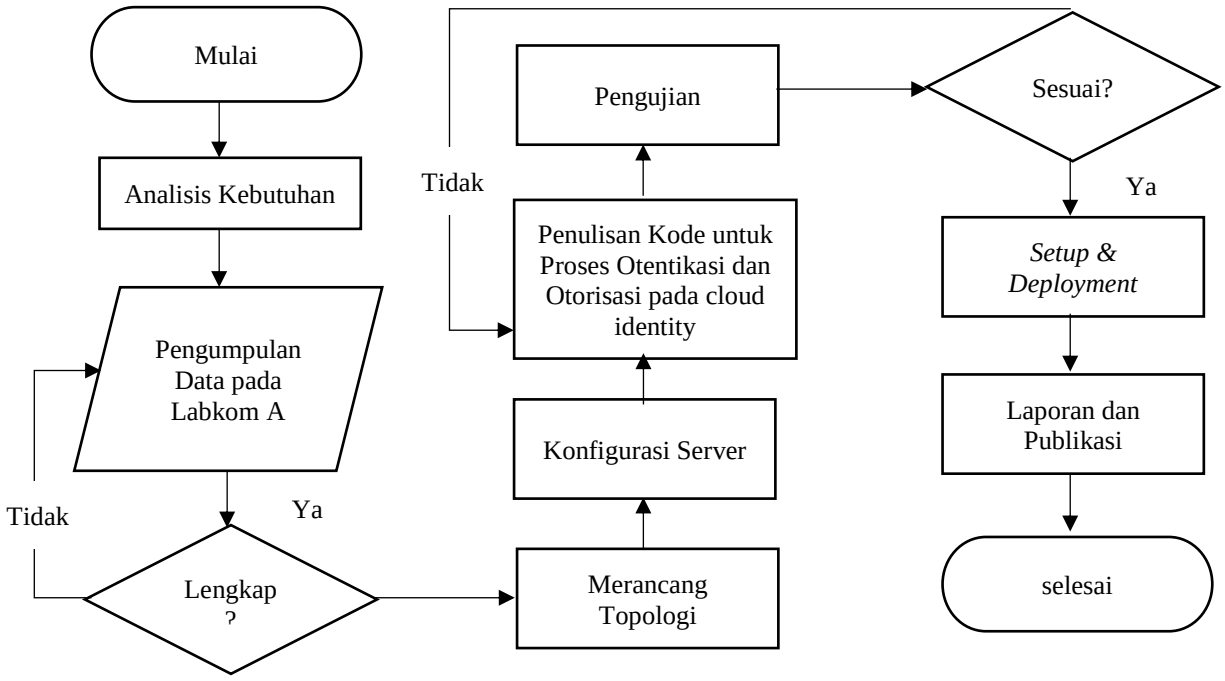
Gambar 2. Ilustrasi Auntentikasi dan Otorisasi

Saat pengunjung tersebut ingin menggunakan lift untuk menuju kamar tersebut, langkah pertama ialah menggunakan kartu untuk mengkases lift. Inilah yang disebut sebagai proses otentikasi. Setelah kartu telah digunakan di lift maka otomatis hanya lantai 7 yang bisa dikunjungi oleh pengunjung tersebut. Inilah yang disebut sebagai proses otorisasi.

III. METODE PENELITIAN

A. *Model Penelitian*

Model penelitian yang dilakukan dalam penelitian ini meliputi tahapan-tahapan yang tampak pada *Flowchart* sebagai berikut:



Gambar 3. Diagram *Flowchart* Penelitian

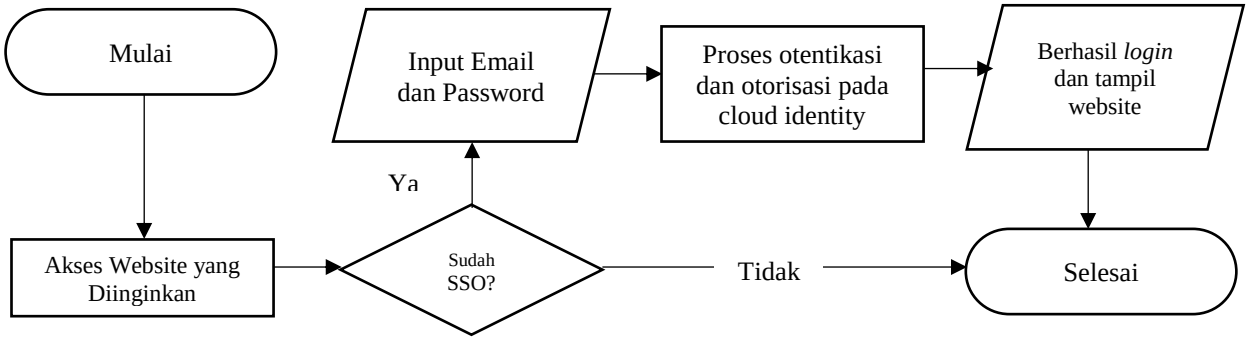
DOI: https://doi.org/10.35457/antivirus.v15i2.1813

Flowchart pada Gambar 3 dapat dijabarkan sebagai berikut:

1) Melakukan kegiatan analisis kebutuhan yang diperlukan dalam pengembangan sistem. Kegiatan ini meliputi identifikasi hardware berupa spesifikasi *server* yang akan digunakan.
2) Melakukan kegiatan pengumpulan data yang relevan terkait pengembangan sistem agar waktu pembangunan menjadi efektif dan efisien.
3) Melakukan perancangan topologi sistem.
4) Melakukan konfigurasi *server*.
5) Tahapan berikutnya melakukan penulisan kode untuk proses otentikasi dan otorisasi.
6) Tahap berikutnya melakukan pengujian kinerja sistem, jika sesuai atau berhasil maka akan dilanjutkan ke tahap berikutnya. Jika belum sesuai, maka kembali ke tahap perancangan dan tahap pengkodean perangkat lunak untuk diperbaiki sampai berhasil.
7) Selanjutnya perangkat lunak yang sudah diuji dan berhasil dilakukan *Setup & deployment* ke server.
8) Tahapan terakhir yaitu membuat laporan dan pembuatan *paper* untuk publikasi sesuai perencanaan yang telah dibuat.

B. *Flowchart* SSO

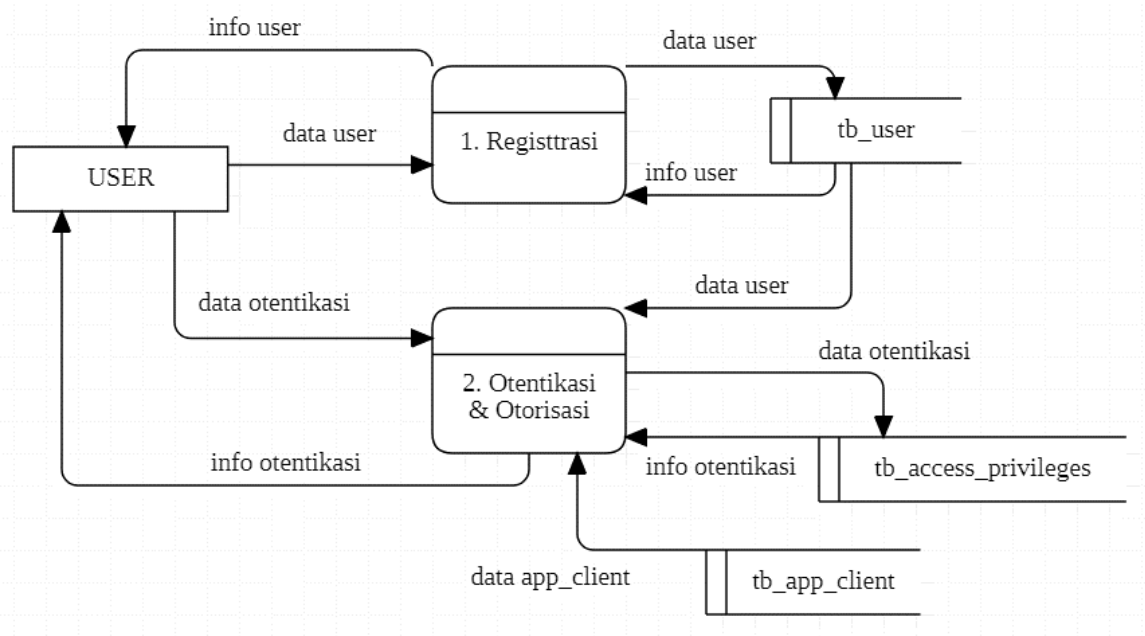
Sistem SSO yang dibangun tampak seperti pada *flowchart* berikut:



Gambar 4. *Flowchart* Sistem SSO

C. *Data Flow Diagram* (DFD) SSO

Sistem SSO yang dibangun tampak pada perancangan DFD seperti berikut:

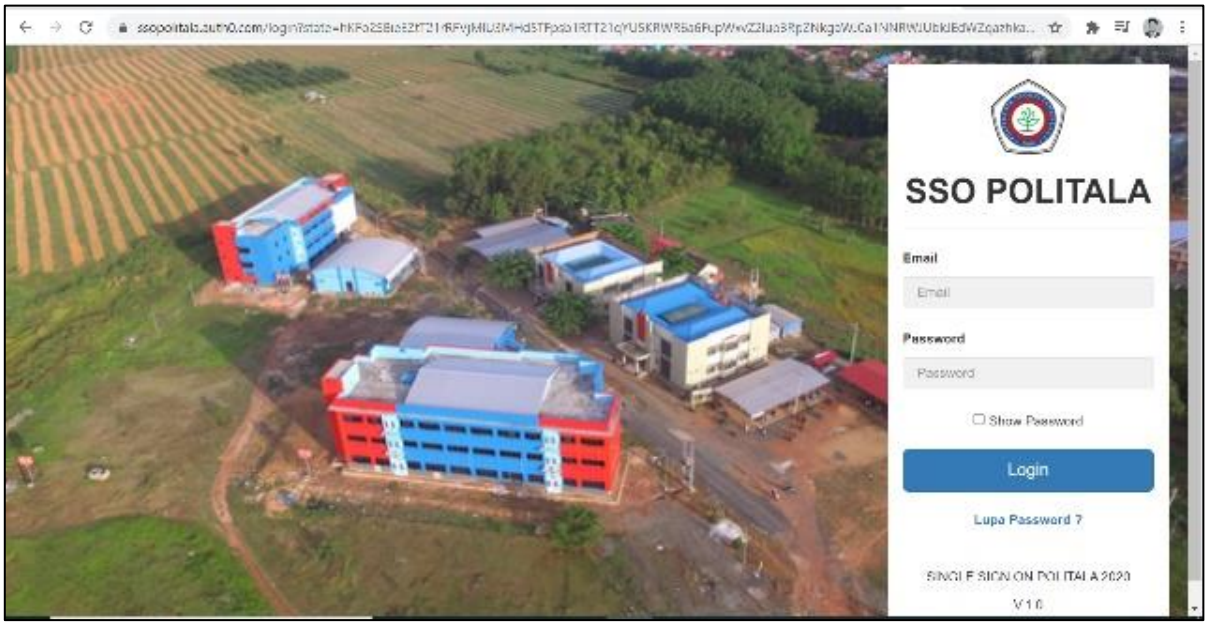


Gambar 5. Data Flow Diagram (DFD) SSO

DOI: https://doi.org/10.35457/antivirus.v15i2.1813

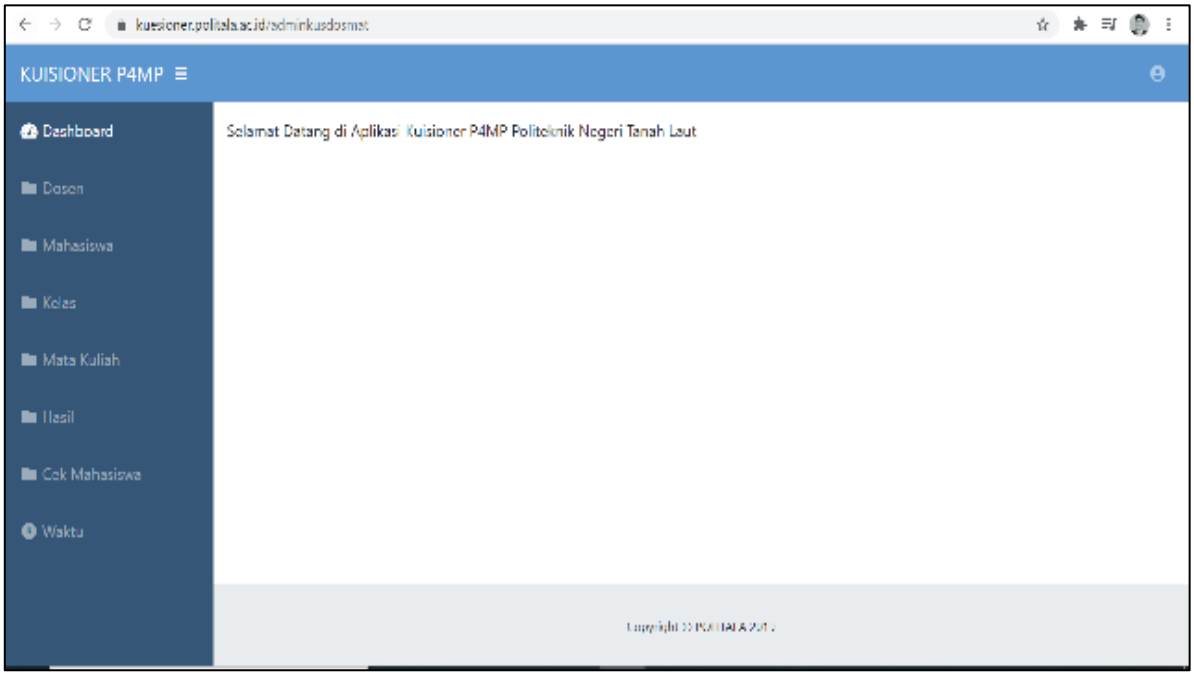
IV. HASIL DAN PEMBAHASAN

Penelitian telah berhasil dilakukan dan diimplementasikan di pada *website* yang dimiliki oleh Politala salah satunya https://kuesioner.politala.ac.id/. Penerapannya data akun pengguna diambil dari Cloud Identity melalui *Secure* LDAP, kemudian data pengguna dikelola oleh RADIUS Server dan didistribusikan ke sistem layanan aplikasi yang ada salah satunya kuisioner Politala. Untuk Cloud Identity sendiri peneliti memanfaatkan fitur yang disediakan oleh Google. Politala sendiri memiliki beberapa *website* seperti https://kuesioner.politala.ac.id/, https://sipadu.politala.ac.id/, https://politala.ac.id/, https://aset.politala.ac.id/ dan lain-lain. Berikut tampilan SSO yang dibangun pada penerapan *website* kuisioner:



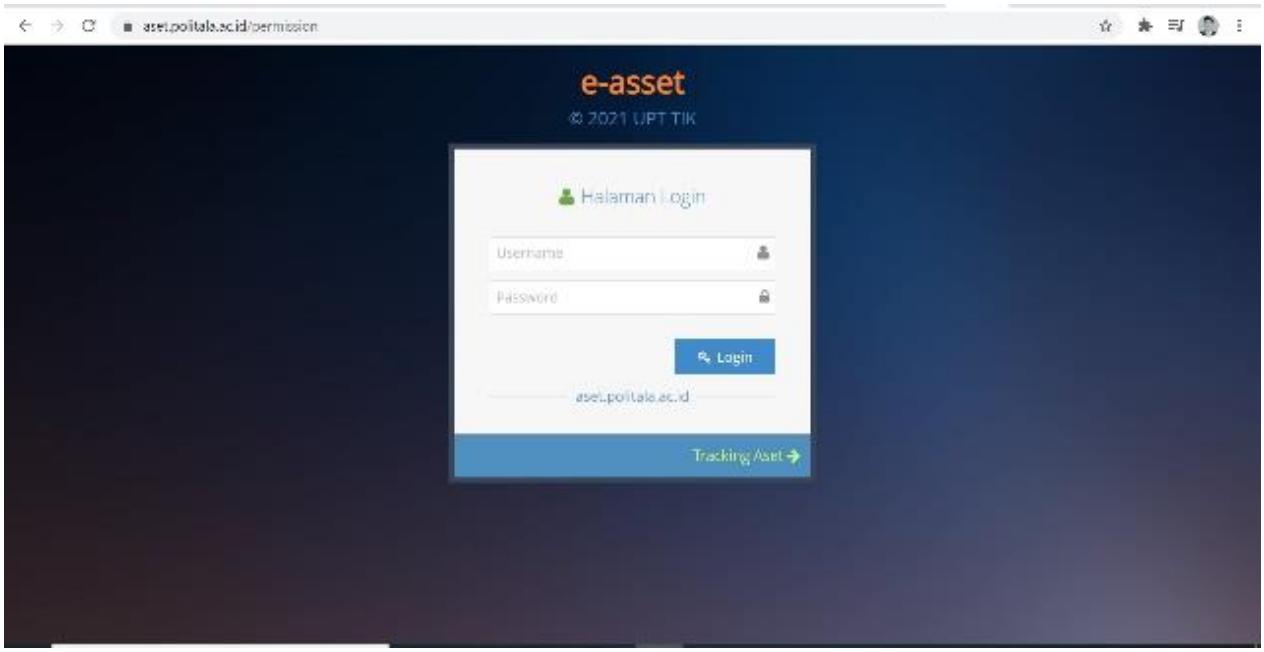
Gambar 6. Tampilan Hasil Penelitian

Pada Gambar 6 tampil halaman SSO setelah melakukan akses pada https://kuesioner.politala.ac.id/. Halaman SSO tersebut terletak pada url https://ssopolitala.auth0.com/login?state. Halaman SSO ini menjadi media untuk dapat mengakses *website* yang dituju dengan mengisikan *E-mail* dan *Password login* pengguna. Jika *E-mail* dan *Password* yang diisikan benar maka akan dialihkan ke *website* yang dituju seperti tampak pada Gambar 7.



Gambar 7. Tampilan Hasil Login

DOI: https://doi.org/10.35457/antivirus.v15i2.1813



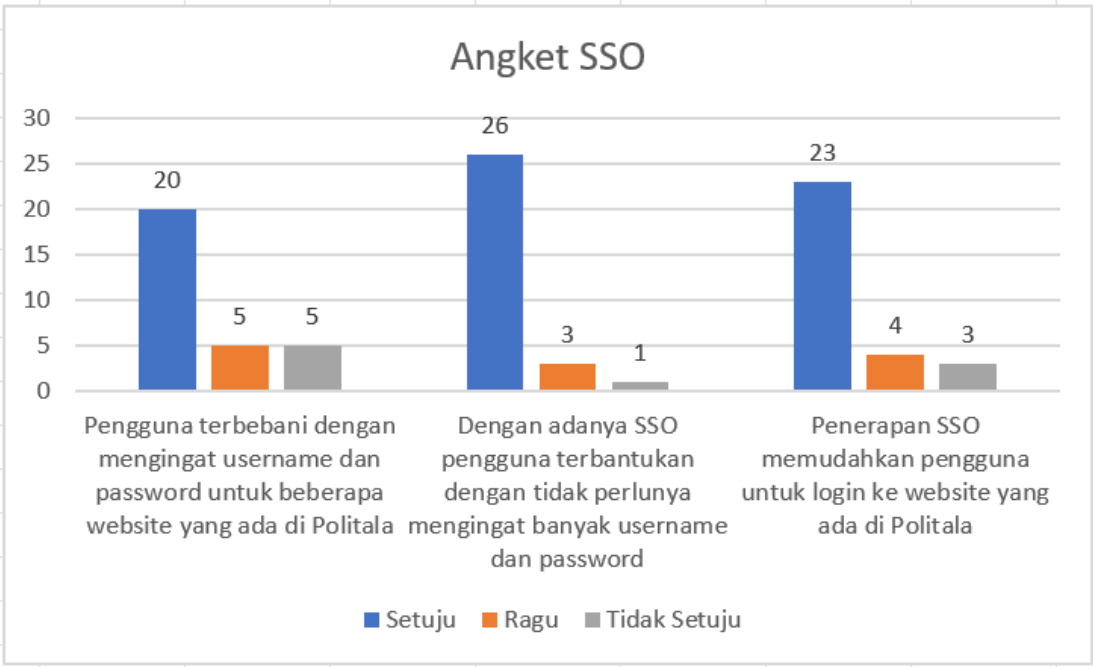
Gambar 8. Tampilan *Website* Tanpa SSO

Gambar 8 merupakan salah satu *website* yang belum dilakukan konfigurasi dengan SSO sehingga *Username* dan *Password* login pada SSO tidak bisa digunakan pada *website* ini. Dengan penerapan SSO akan mempermudah pengguna dalam mengakses *website* tersebut yaitu hanya menggunakan sebuah *Username* dan *Password* untuk berbagai *website* yang telah dikonfigurasi. Hal ini untuk menghindari pengguna dalam memliki *Username* dan *Password* yang banyak dalam sebuah sistem yang telah terintegrasi.

Uji validasi dilakukan dengan menyebarkan angket terhadap 30 orang pengguna untuk mendapatkan respon terhadap SSO yang telah diterapkan. Pernyataan yang diajukan meliputi:

1) Pengguna terbebani dengan mengingat *Username* dan *Password* untuk beberapa *website* yang ada di Politala
2) Dengan adanya SSO pengguna terbantukan dengan tidak perlunya mengingat banyak *Username* dan *password*
3) Penerapan SSO memudahkan pengguna untuk login ke *website* yang ada di Politala.

Pilihan penilaian angket terdiri dari 3 pilihan yang terdiri dari Setuju, Ragu dan Tidak Setuju dan hasil dari angket tersebut tampil pada Gambar 9.



Gambar 9. Hasil Uji Validasi

DOI: https://doi.org/10.35457/antivirus.v15i2.1813

Hasil dari uji validitas diatas tampak bahwa jumlah respon tertinggi pada pilihan Setuju ialah pada pernyataan bahwa pengguna dimudahkan dengan adanya SSO yaitu dengan tidak lagi mengingat banyak *Username* dan *Password* untuk beberapa *website* yang ada di Politala

V. KESIMPULAN

Kesimpulan dalam penelitian ini ialah telah berhasil dibangun dan diterapkan SSO dengan memanfaatkan Cloud Identity sebagai sumber data pengguna. Hal ini untuk mempermudah pengguna dalam login ke beberapa sistem yang terintegrasi cukup hanya menggunakan sebuah *Username* dan *Password*. Pengembangan selanjutnya bisa dilakukan dengan pengembangan otorisasi dan verifikasi login dengan akun Gmail/Google yang dimiliki pengguna. Hal ini akan sangat mempermudah sebab hampir semua pengguna internet saat ini memiliki akun Google.

UCAPAN TERIMA KASIH

Ucapan terima kasih disampaikan kepada Politeknik Negeri Tanah Laut atas pendanaan penelitian yang diberikan dengan kontrak 023/PL-40.5/LT/2021 tahun 2021.

DAFTAR PUSTAKA

[1] J. De Clercq, "Single sign-on architectures," in *International Conference on Infrastructure Security*, 2002, pp. 40–58.
[2] H. Yuliansyah, "Dan Otorisasi Untuk Proses Login Multi Aplikasi Web," *Semin. Nas. Inform. 2011*, vol. 2011, no. semnasIF, pp. 17–23, 2011.
[3] A. H. Muttaqin, A. F. Rochim, and E. D. Widianto, "Sistem Otentikasi Hotspot Menggunakan LDAP dan RADIUS pada Jaringan Internet Wireless Prodi Teknik Sistem Komputer," *J. Teknol. dan Sist. Komput.*, vol. 4, no. 2, p. 282, 2016, doi: 10.14710/jtsiskom.4.2.2016.282-288.
[4] S. Qidri, M. Asfi, R. Taufiq, and M. Hatta, "Pengelolaan Hak Akses *User* Jaringan Menggunakan FreeRADIUS Untuk Login Jaringan," *J. Sains dan Inform.*, vol. 6, no. 2, pp. 183–192, 2020.
[5] G. Guntoro and M. Fikri, "Perancangan Aplikasi Single Sign-On Menggunakan Otentikasi Gambar," *Digit. Zo. J. Teknol. Inf. dan Komun.*, vol. 9, no. 1, pp. 12–21, 2018, doi: 10.31849/digitalzone.v9i1.648.
[6] LDAP, "LDAP," 2021. https://ldap.com/ (accessed Oct. 01, 2021).
[7] R. C. Satriawan, "PENGEMBANGAN SISTEM OTENTIKASI SINGLE SIGN ON MENGGUNAKAN PROTOCOL LDAP (LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL)." University of Muhammadiyah Malang, 2017.
[8] R. Ramadhan and D. A. Kurnia, "Otentikasi *User* Secara Terpusat Menggunakan FreeRADIUS Dalam Upaya Mengoptimalkan Jaringan Hotspot," *J. ICT Inf. Commun. Technol.*, vol. 15, no. 1, pp. 17–22, 2016.
[9] FreeRADIUS, "FreeRADIUS," 2021. https://freeRADIUS.org/ (accessed Oct. 01, 2021).
[10] R. Sandhu and P. Samarati, "Authentication, access control, and audit," *ACM Comput. Surv.*, vol. 28, no. 1, pp. 241–243, 1996.

DOI: https://doi.org/10.35457/antivirus.v15i2.1813