Sistem Keamanan SSO Pada Jalur Komunikasi Berbasis SAML Menggunakan Digital Signature

Frisilia Indahni[1], Yesi Novaria Kunang[2], Ari Muzakir[3]
[1,3] Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Bina Darma
[2] Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Bina Darma
Palembang, Indonesia
[1] indahni.frisilia@yahoo.com, [2] yesi_kunang@mail.binadarma.ac.id

Abstract. Single sign on merupakan teknologi yang mengizinkan pengguna untuk melakukan otentikasi pada beberapa aplikasi web hanya menggunakan satu username dan satu password. Pengguna cukup melakukan login sekali agar bisa mengakses beberapa aplikasi web yang terintegrasi. Single sign on menyediakan fasilitas Security Assertion Markup Language (SAML) sebagai portal penghubung antara pengguna dan aplikasi web. Dengan menggunakan beberapa aplikasi web yaitu moodle dan wordpress. SSO SAML menggunakan digital signature sebagai sistem keamanan antar server dengan menggunakan sertifikat SP, sertifikat Idp dan sertifikat CAS. Digital signature memiliki fungsi sebagai penanda pada data yang memastikan bahwa data tersebut adalah data yang sebenarnya (tidak ada yang berubah) dengan menggunakan algoritma RSA .

Keywords: Single Sign On (SSO), Security Assertion Markup Language (SAML), Digital Signature.

1 Pendahuluan

Kemajuan dan perkembangan teknologi yang semakin cepat dan mudah telah berpengaruh pada hampir seluruh kehidupan manusia, tidak terkecuali kebutuhan akan informasi dan komunikasi yaitu internet. Aplikasi web sekarang ini banyak digunakan seperti *blog* dan *e-learning*, untuk memudahkan integrasi aplikasi *web* maka digunakan sistem *single sign on* (SSO). SSO adalah sebuah sistem yang mengizinkan pengguna untuk menggunakan layanan dari berbagai situs *web* dalam bermacam-macam lingkungan layanan aplikasi *web* yang berbeda, tanpa *credential* (ID dan *password*) jika pengguna telah melakukan otentikasi pada salah satu situs *web*. Oleh karena itu, SSO merupakan sebuah solusi yang menawarkan kenyamanan, efisiensi dan keamanan yang tinggi bagi pengguna dan pengelola dalam mengakses berbagai layanan aplikasi web [1].

Agar memudahkan *user* dalam pengaksesan *web*, bahwa ketika *user* mengakses aplikasi yang telah di-*authorize* untuk diakses maka permintaan autentikasi dari *user* akan dihilangkan ketika *user* membuka aplikasi yang lain. Sehingga memudahkan proses *login* dengan sekali *login* saja melalui *web* portal.

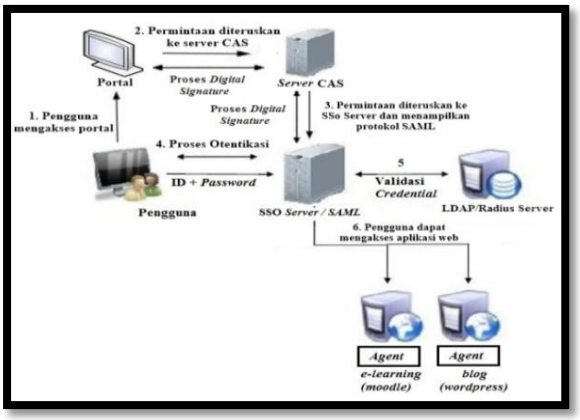
SSO juga membutuhkan sistem keamanan agar data dapat terjaga dengan aman, ada beberapa macam framework yang menyediakan sistem keamanan *web single sign on* salah satunya yaitu *security assertion markup language* (SAML). SAML menyediakan, solusi XMLbased aman untuk bertukar informasi keamanan pengguna antara *identity provider* dan *service provider* (ASP atau SaaSs) [2]. Standar SAML mendefinisikan aturan dan sintaks untuk pertukaran data, namun fleksibel dan dapat memungkinkan untuk data kustom untuk ditransmisikan ke penyedia layanan eksternal. Namun, SAML tidak menjamin akan kerahasian dan keaslian data, sehingga dibutuhkan sistem keamanan antar aplikasi salah satunya dengan menggunakan kriptografi kunci publik khususnya Digital Signature yaitu Digital Signature adalah suatu nilai kriptografi yang bergantung pada pesan dan pengirim pesan/signer.

Namun algoritma kriptografi untuk membuat *digital signature* misalnya *digital signature algorithm* (DSA), *rivest, shamir, andleman* (RSA), atau *elliptic curve digital signature algoritm* (ECDSA) hanya menghasilkan satu *digital signature* untuk satu *e-document*. Hal ini tidak sesuai dengan konsep *digital signature* yang bergantung pada pengirim/*signer* dimana signer lebih dari satu. Sehingga perlu konsep baru mengenai *digital signature* yang dapat berfungsi sebagai otorisasi suatu e-dokumen sama halnya otorisasi tanda tangan (*handwritten*) beberapa signer pada dokumen fisik. Salah satu *signer* adalah dengan *biometric* tanda tangan *offline*. Tanda tangan *offline* adalah tanda tangan pada dokumen fisik yang digitasi oleh scanner [3]. *Digital Signature* memungkinkan penerima informasi untuk menguji terlebih dahulu keaslian informasi yang didapat dan meyakinkan bahwa data yang diterima itu dalam keadaan utuh.

Penelitian ini melibatkan aplikasi *web* berupa *blog* dan *e-learning*. *Blog* sebagai media untuk menyebarluaskan pengetahuan melalui *internet* [4]. Sedangkan *E-learning* menawarkan konsep belajar menjadi *placeless*, *boarderless* dan *timeless* [5]. *Moodle* adalah sebuah aplikasi web gratis yang pendidik dapat digunakan untuk membuat situs pembelajaran *online* yang efektif.

2 Metode Penelitian

Metode penelitian yang digunakan adalah metode penelitian tindakan (*action research*) [6], adapun tahapan-tahapannya sebagai berikut: 1) Mendiagnosa (*diagnosing*), 2) Melakukan perencanaan tindakan (*action planning*), 3) Melakukan evaluasi (*evaluating*), dan 4) Menentukan pembelajaran dari hasil penelitian (*learning*).



Gambar 1. Perancangan

Dari gambar di atas dapat dilihat bahwa awalnya pengguna mengakses portal, akan ada proses permintaan sertifikat *digital signature* selanjutnya diteruskan ke login SSO untuk mengaktifkan protokol SAML. Pada *form login* pengguna akan mengisi *credential* (*username* dan *password*) untuk kemudian akan dilakukan pengecekan validasi *credential* di *server radius*. Ketika pengguna berhasil *login*, maka secara otomatis aplikasi web dapat diakses tanpa perlu melakukan login kembali.

3 Hasil dan Pembahasan

Konfigurasi server dan client menggunakan local.domain, sehingga perlu dilakukan setting IPAddress pada masing-masing komputer, agar server dan client dapat berkomunikasi untuk membentuk sistem otentikasi SSO dengan SAML. Awalnya pengguna mengakses portal dan akan di redirectdari sertifikat SP ke sertifikat Idp akan terjadi pertukaran key public SP dan *private key* IdP untuk proses enkripsi dan penandatanganan digital setelah shibboleth berhasil selanjutnya otentikasi shibboleth-sp dan shibboleth idp menggunakan kunci public dari sertifikat CAS untuk menuju ke server CAS. Dari server CAS, pengguna akan diminta memasukkan *username* dan *password* untuk validasi user pada server radius/ldap. Setelah pengguna berhasil login, maka secara otomatis aplikasi web dapat diakses tanpa perlu melakukan login kembali.

3.1 Konfigurasi *Shibboleth*

Shibboleth adalah bagian dari SAML yang merupakan sistem keamanan otentikasi yang digunakan pada sistem SSO yangd dibuat. SAML terdiri dari Idp dan SP yang merupakan penyedia identitas dan layanan, dimana akan terjadi otentikasi dan

otorisasi di dalam proses nya. Instalasi dan konfigurasi shibboleth, yaitu: 1) Shibboleth Identity Provider (IdP). IdP adalah penyedia layanan identitas, yang mana di sini akan mengecek identitas user yang login, apakah sesuai atau tidak. Softwareshibboleth Idp bisa di download di http://shibboleth.net/downloads/ yang kami gunakan adalah shibboleth-1.3.2.tar.gz. Berikut adalah proses dalam instalasi Idp, dan 2) *Shibboleth Service Provider* (SP). SP adalah penyedia layanan, dimana sebelum tahapan ke Idp, terlebih dahulu akan mengalami proses di SP untuk menyediakan layanan *web*. *Software* SP bisa di-*download* di www.shibboleth.net/downloads/service-provider yang digunakan adalah shibboleth-1.3-11.src.rpm.

3.1 Konfigurasi *Digital Signature*

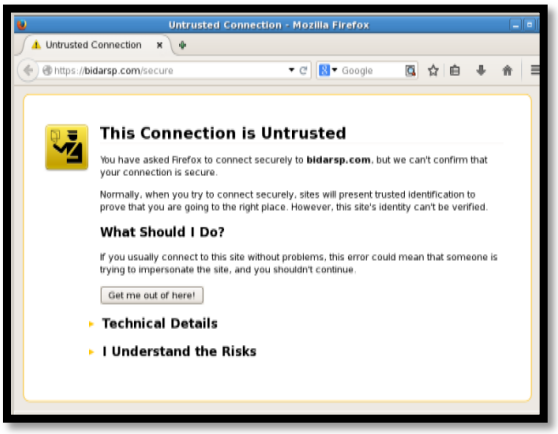
Langkah-langkah untuk konfigurasi *digital signature*, adalah: 1) Lakukan *install apache* untuk mengaktifkan mod_ssl yang digunakan untuk layanan HTTP, 2) Membuat sertifikat SP dengan *generate* kunci menggunkan RSA 2048. Lalu buat sertifikat Idp dengan konfigurasi yang sama, 3) Membuat *certificate signing request* (CSR). Setelah terbentuk kunci privat, selanjutnya membuat CSR. Idealnya, CSR akan dikrimkan ke sebuah CA, kemudian mereka akan melakukan verifikasi terhadap identitas dari pihak yang memintasertifikat tersebut untuk ditanda tangani. Karena tidak memiliki sertifikat yang dapat ditanda tangani oleh CA tersebut, maka opsi yang lain adalah dengan *self-sign* CSR. File CSR dibuat pada sertifikat idp dan sp dengan konfigurasi yang sama, dan 4) Menandatangani Sertifikat. Hal selanjutnya yang harus dilakukan adalah menandatangani sertifikat yang telah dibuat tadi dengan kunci privat yang dibuat sendiri. Karena itulah cara ini disebut *self-sign certificate*.

3.3 Pengujian *Shibboleth*

Pengujian shibboleth dengan mengakses alamat URL http://bidar.com, akan muncul tampilan portal login shibboleth.

3.4 Pengujian *Digital Signature*

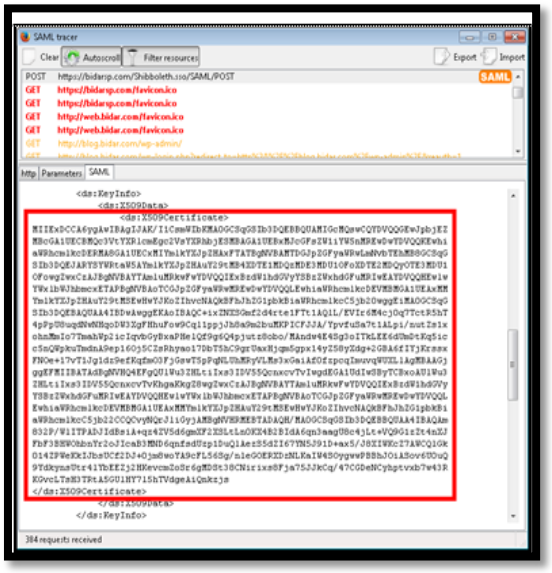
Langkah berikutnya adalah melakukan pengujian dengan mengunjungi portal shibboleth http://bidar.com dan akan tampil sertifikat sp dan idp.



Gambar 2. Sertifikat SP

Pilih "I understand the Risk" (gambar 2) dan percaya bahwa ini adalah situs yang aman, bukan alamat *phising*. Sertifikat sp akan me-redirect permintaan ke alamat https://bidaridp.com. Setelah melewati proses enksripsi pada bidarsp.com akan ada permintaan dari sertifikat idp, pada proses ini sertifikat idp mendeskripsikan pesan menggunakan kunci privat pasangan dari kunci publik dan akan terjadi penandatangan digital, sertifikat inilah yang menjamin keaslian data dari *user*.

Kemudian akan tampil sertifikat http://sso.bidar.com:8443/cas pada tahap ini akan ada proses otentikasi user.



Gambar 3. SAML *Tracer*

Sertifikat CAS dikonfigurasi menggunakan portecle dengan kunci publik. Sertifikat ini berfungsi sebagai validasi dan mengontentikasi permintaan user ke

portal CAS mengunakan kunci publik dan sebagai bukti koneksi yang sah antar server dan client. Untuk mengetahui sertifikat yang sudah terenkrip dapat dilihat pada *SAML tracer* (gambar 3) yang sudah diaktifkan sejak mengakses portal http://bidar.com.

Sertifikat diatas merupakan sertifikat yang sudah tertandatanganni. <KeyInfo> elemen yang memungkinkan penandatanganan sertifikat digital X.509 dengan menggunakan kunci yang memvalidasi tanda tangan biasanya dibuat dengan kunci asimetri RSA dan untuk mengenkripsi menggunakan SHA1.

4 Kesimpulan

Berdasarkan hasil ujicoba dengan melakukan analisis menggunakan penyadapan (*sniffing*) terhadap komunikasi data antara *client* dan *server*, maka dapat disimpulkan:

1. Sistem keamanan *digital signature* dapat digunakan sebagai otentikasi sumber pesan yang menjamin keaslian data agar terhindar dari penyangkalan oleh pihak ketiga.
2. SAML dapat dihubungkan pada CAS dengan cara otentikasi SSO pada saat verifikasi *digital signature* hasil *redirect* dari bidarsp sebagai *Service Provider* (SP) dan bidaridp sebagai *Identity Provider* (IdP) yang akan diterima oleh sertifikat CAS.
3. *Digital Signature* memanfaatkan 3 sertifikat untuk menjamin otentikasi dan *integrity* yaitu sertifikat SP yang akan mengenkripsi jalur komunikasi, sertifikat IdP akan mengverifikasi penandatanganan pada pesan yang sudah dienkripsi dan sertifikat CAS yang akan mengotentikasi.
4. Dari hasil pengujian diketahui bahwa *digital signature* mengamankan dari jalur komunikasi portal SAML sampai login CAS, sedangkan pada jalur aplikasi *header* tidak terenkripsi tetapi *content* aplikasi terenkrip.

Daftar Pustaka

1. Nursyamsi, "Implementasi Sistem Single Sign-On Berbasis Java," Sarjana Teknik, Departemen Teknik Elektro, Universitas Sumatra Utara, Medan, 2009.
2. K. D. Lewis and J. E. Lewis, "Web single sign-on authentication using SAML," *International Journal of Computer Science Issues (IJCSI),* vol. 2, 2009.
3. R. Munir, *Kriptografi*. Bandung: Informatika, 2006.
4. L. A. Abdillah, "Managing information and knowledge sharing cultures in higher educations institutions," in *The 11th International Research Conference on Quality, Innovation, and Knowledge Management (QIK2014)*, The Trans Luxury Hotel, Bandung, Indonesia, 2014.
5. L. A. Abdillah, "Students learning center strategy based on e-learning and blogs," in *Seminar Nasional Sains dan Teknologi (SNST) ke-4 Tahun 2013*, Fakultas Teknik Universitas Wahid Hasyim Semarang 2013, pp. F.3.15-20.
6. S. Madya, *Teori dan Praktik Penelitian Tindakan*. Bandung, 2006.