INFOMATEK
Volume 21 Nomor 2 Desember 2019

PENETAPAN KONTROL KEAMANAN SISTEM TERPADU UNPAS (SITU) BERDASARKAN KONSEP SECURE SOFTWARE DEVELOPMENT LIFE DAN HASIL UJI PENETRASI

Rita Rijayanti[*)], Iwan Kurniawan

Program Teknik Informatika
Fakultas Teknik – Universitas Pasundan

Abstrak: Pemanfaatan teknologi dapat memunculkan usaha mendapatkan keuntungan yang dapat merugikan perorangan ataupun organisasi, fenomena ini juga terjadi pada Sistem Informasi Terpadu UNPAS (SITU), dimana sistem mengalami serangan yang menyebabkan kerugian layanan secara keseluruhan menjadi terganggu, sehingga dibutuhkan sebuah kontrol keamanan yang dapat menunjang kelancaran dari sistem berjalan. Proses penetapan kontrol keamanan yang dibuat mengacu pada konsep *Secure Software Development Life Cycle* (SSDLC), yang diawali dengan proses analisi hasil uji penetrasi, pemetaan tingkat keamanan berdasarkan konsep management risiko dan dikaitkan dengan kondisi lingkungan sistem, dan yang menjadi parameter penentu kontrol keamanan adalah ancaman dan keandalan dari sistem berjalan berdasarkan hasil dari uji penetrasi pada penelitian sebelumnya "Penentuan Performansi Sistem Informasi Terpadu Unpas (SITU) melalui Uji Penetrasi.

Kata kunci: Kontrol Keamanan, Management Risiko, *Secure Software Life Cycle*

I. PENDAHULUAN

Universitas Pasundan (UNPAS) merupakan organisasi jasa pendidikan yang cukup besar dan berkembang di wilayah Jawa Barat, dengan nama besar tersebut UNPAS mencoba memberikan pelayanan terbaik kepada para pengguna jasa mereka. Salah satu penunjang pelayanannya adalah dengan memanfaatkan teknologi informasi dalam pengelolaan data dan informasi untuk kegiatan operasional keseharian seperti: pendaftaran dan seleksi mahasiswa baru, kegiatan akademik, keuangan, penjadualan, kehadiran, penilaian dan sebagainya. Dengan pemanfaatan teknologi tersebut, memunculkan usaha yang diduga sengaja dilakukan oleh orang-orang yang tidak bertanggung jawab untuk merugikan organisasi, baik secara finasial maupun kepercayaan dengan melakukan penyerangan-penyerangan terhadap sistem berjalan. Data dalam jaringan informasi merupakan objek yang riskan terintervensi

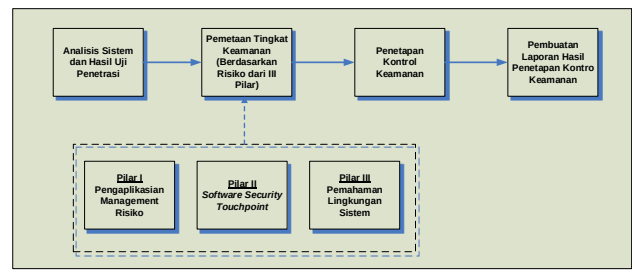
[*)] rita.rijayanti@unpas.ac.id

Pertama diterima : 29 Agustus 2019
Direvisi : 27 September 2019
Disetujui untuk publikasi: 29 September 2019

oleh peretas (Akhirina dkk. [1]). Melihat fenomena tersebut dapat dilihat perlunya melakukan penelitian untuk membuat kontrol keamanan bagi Sistem Informasi Terpadu UNPAS (SITU) melanjutkan penelitian sebelumnya mengenai penentuan performansi sistem informasi terpadu UNPAS (SITU) melalui uji penetrasi dimana hasil dari penelitian sebelumnya menghasilkan banyak celah keamanan yang diakibatkan oleh vurneability.

II. METODOLOGI

Metode yang akan digunakan pada penelitian ini adalah didasarkan pada konsep *Secure Software Development Life Cycle* dalam penetapan kontrol keamanan sistem. Dimana tahapan awal dari penelitian dapat dilakukan mulai dari melakukan analisis sistem dan mempelajari hasil uji penetrasi dari penelitian sebelumnya, yang dilanjutkan dengan melakukan pemetaan tingkat keamanan dengan mengacu pada perinsipprinsip tiga pilar yaitu, penerapan management risiko, konsep *Software Secure Touchpoint* dan pemahaman lingkungan sistem. Dimana ketiga tahap tersebut nantinya akan menghasilkan sebuah bentuk rancangan mitigasi risiko yang dapat digunakan untuk menetapkan control keamanan apa saja yang dapat diterapkan. pada sebuah sistem, dan sampai dengan akhirnya pembuatan laporan hasil penetapan kontrol keamanan bagi Sistem terpadu Unpas (SITU) (Pressman [2]), (Hendayun [3]).



Gambar 1.

Metode Penelitian

Uji penetrasi terhadap jaringan merupakan metode yang terbaik untuk menilai tingkat keamanannya (Pujiarto dkk. [4]), (Dirgahayu dkk. [5]).

III. ANALISIS DAN PEMBAHASAN

3.1 Analisis Sistem Berjalan

Analisis sistem berjalan ini dimaksudkan untuk memberikan gambaran sistem yang sedang berjalan di Fakultas Teknik Universitas Pasundan, sistem yang diamati adalah sistem akademik secara keseluruhan, mulai dari KRS, jadual perkuliaan, bimbingan, nilai dan sebagainya. Berikut adalah beberapa hal yang dibutuhkan dalam proses analisis guna menunjang dalam penentuan kerentanan/ potensi masalah yang akan menyebabkan kelemahan sebuah sistem.

1. Spesifikasi Environment server situ Akademik

   Dari sisi spesifikasi environment server terbagi menjadi dua bagian yaitu sisi Software dan Hardware, dengan detail terlihat pada Tabel 1.

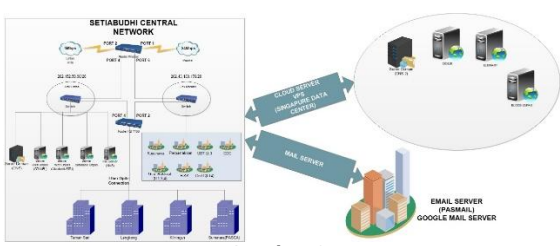
Tabel 1.

Kebutuhan SW dan HW

Software	*Hardware*
1. XenServer Virtualization Hypervisor	1. IBM Server X3650
2. Apache 2.2	2. Intel Xeon 6 Core 2.9Ghz
3. Mysql 5.6	3. RAM 16GB
4. PHP 5.6	4. Disk 150GB
5. Virtualmin Control Panel	
6. WildCard SSL Comodo	

2. Topologi Interkoneksi Antar Server

   Berikut adalah gambaran topologi yang dimiliki oleh Fakultas Teknik Universitas Pasundan dalam mengelola sistem informasi terpadunya dapat dilihat pada Gambar 2.



Gambar 2.

Teknologi Interkoneksi Antar Server

3. Dokumentasi Basis Data (skema), terdiri atas:

   Informasi Umum Mhs

   · Layanan Dekanat
   · Layanan Prodi
   · Layanan Dosen
   · Layanan Mhs
   · Layanan SBAP
   · Dikti
   · Setup dan Master Data
   · Keuangan
   · Registrasi
   · Perkuliahan Nilai
   · Kelulusan

4. Dokumen Hak Akses, terdiri atas:

   · Dkn
   · PD I
   · PD II
   · PD III
   · KDBAP/SBAP FT1
   · Staf SBAP
   · Staf DHMD
   · KSB. Keu
   · Kajur/Sekjur/Dikjar
   · Adm. Nilai Jur
   · Adm. TA Jur.
   · Adm. KP Jur
   · Adm Umum Jur
   · Dosen
   · Mhs

Hasil Scanning menggunakan Acunetix http://akademik.unpas.ac.id/ft/besan.depan.php, yang didapat beberapa celah keamanan yang muncul dan di antaranya ada yang dapat mengakibatkan sampai dengan sistem terhenti, berikut berdasarkan hasil pengimplementasian menggunakan beberapa tools vulnerability dapat dilihat pada Tabel 2.

Tabel 2.

Daftar Vulnerability

Low	
1.	Clickjaking: X-Frame-Options header missing
2.	Possible sensitive directories
3.	Possible virtual host found
4.	Session Cooki without HttpOnly flag set
5.	Session Cooki without secure flag set
6.	Slow response time
Medium	
1.	Application error message
2.	HTML from without CSRF protection
3.	Password field submitted using GET method
4.	Slow HTTP Denial of Service Attack
5.	Source code disclosure
6.	Webalizer script
High	
1.	Cross site scripting (verified)
2.	Cross site scripting [stored] (verifird)
3.	Host header attack
4.	SQL injection

3.2 Penilaian Risiko

Penilaian risiko diawali dengan menetapkan semua risiko yang mungkin muncul berdasarkan hasil uji pentrasi, namun pada penelitian ini hanya akan dikhususkan untuk kasus-kasus dengan nilai *High* (lihat Table 3). Setelah seluruh risiko yang mungkin muncul dianalisis, kemudian dilihat frekuensi kemunculan dan dampak yang muncul, sehingga dapat ditetapkan nilai setiap risiko yang ada, dengan menetapkan nilai Low (L), Medium (M) dan High (H). Berikut adalah daftar hasil penilaian risiko yang sudah dilakukan tim seperti yang terlihat pada Tabel 3.

Tabel 3

Penilaian Risiko

No	Risiko	Frek. Kemunculan	Tingkat Risiko
1	Kehilangan Data Sensitif	30	H
2	Kehilangan Source Code / perusakan soruce code	30	H
3	Pembajakan session	25	H
4	Deface	1	H
5	Permasalahan perfomansi perangkat lunak.	449	H

3.3 Rekomendasi Kontrol Keamanan

Pada tahapan ini akan ditetapkan kontrol-kontrol keamanan yang akan dibuat, didasarkan pada hasil penilaian risiko, literatur dan kondisi organisasi seperti terlihat pada Tabel 4.

Tabel 4

Rekomendasi Kontrol Keamanan

No	Risiko	Rekomendasi Kontrol Keamanan

1	Kehilangan data sensitif	- Adanya aturan bahwa setiap aktivitas atau perubahan yang berkaitan dengan data sensitif harus tercatat (log).
		- Backup data dibuat terjadual secara realtime.
		- Adanya pengontrolan dalam pembuatan script codingan:
		Script PHP yang memiliki Vulnerable terhadap bug XSS :
		<?php
		$komentar = $_POST['komentar'];
		// "<script>alert('bug xss');</script>"
		echo $komentar; // browser memberi sebuah alert/dialog 'bug xss'.?>
		Dengan memastikan selalu

No	Risiko	Rekomendasi Kontrol Keamanan

		menambahkan htmlspecialchars($string, ENT_QUOTES), sehingga kodenya menjadi seperti ini :
		
		<?php $komentar = $_POST['komentar']; // <script>alert('bug xss');
		
		</script>echo htmlspecialchars($komentar, ENT_QUOTES);//browser tidak memberi sebuah alert/dialog 'bug
		
		xss'.//lantas dikonvert menjadi kode HTML entities.?>
		
		- Merubah script php, menggunakan MySQL_escape_string dan melakukan pemfilteran karakter ‘ dengan memodifikasi php.ini
		- Melakukan traching terhadap logika kode yang sudah dibuat dan melakukan pengecekan ulang analisis dan design yang telah dibuat sebelumnya.

2	Kehilangan Source Code / perusakan soruce code	Selalu ada backup data secara realtime.
3	Pembajakan Session	Melakukan review ulang dalam mendesain atau dalam mengimplementasikan mekanisme session tracking. **Catatan:** Tidak ada satupun patch sistem operasi, firewall atau konfigurasi web server yang dapat mencegah serangan Session hijacking, sehingga setiap pengembang web harus mengerjakan secara cermat desain dan implementasi session dan state tracking.
4	Defacce	- Hardening website dan source wajib dilakukan, misalkan jangan menggunakan ”default configuration”, aturlah sedemikian rupa ”configuration website” dengan memperhatikan: permission, acces level, indexiding, database configuration, password dan user management. - Melakukan pengecekan terhadap security update secara berkala. - Melakukan pengontrolan aplikasi dengan melakukan penetration testing terhadap website secara berkala, baik secara lokal maupun langsung di website.
7	Permasalahan perfomansi perangkat lunak.	- Adanya review code yang dibuat oleh Developer atau jika memungkinkan dibuat standard code dalam pengembangan perangkat lunak. - Pemantauan koneksi jaringan dan perangkat-perangkat yang digunakan. - Pemantauan Kapasitas bandwidth (disesuaikan berdasarkan kebutuhan.)

V. KESIMPULAN

Hasil dari penelitian ini perancangan rekomendasi kontrol keamanan TI berdasarkan hasil dari penilaian dan mitigasi risiko dilihat dari hasil vulnerability yang muncul pada kondisi sistem berjalan, dimana permasalahan difokuskan pada tahapan perancangan dan design pengembangan perangkat lunak. Didapat rancangan kontrol yang direkomendasikan berupa tahapan secara deskriptif baik itu dari sisi script, tahapan atau pun aturan dalam pengembangan perangkat lunak berdasarkan konsep Secure Software Development Life Cycle (SSDLC).

DAFTAR PUSTAKA

[1] Akhirina, T.Y., Arif, S.M., Rahmatika. “Evaluasi Keamanan Teknologi Informasi pada PT Indotama Partner Logistics menggunakan Indeks Keamanan Informasi (KAMI),” *Teknosi*, vol. 2 no. 2, pp. 53-62, 2016.

[2] Pressman, R. S. *Software Engineering a Practitioner's Approach*, Seventh Edition. New York: McGrawHill Radack, Shirley. 2010.

[3] Hendayun, M. *Software Secure Engineering.* Materi Perkuliahan. 2011.

[4] Pujiarto, B., Utami, E., Sudarman. “Evaluasi Keamanan Wireless Local Area Network menggunakan Metode Penetration Testing (Kasus: Universitas Muhammadiyah Magelang),” *Jurnal Dasi*, vol. 14 no. 2, pp.16-20, 2013.

[5] Dirgahayu, R.T., Prayudi, Y., Fajaryanto, A. “Penerapan Metode ISSAF dan OWASP versi 4 untuk Uji Kerentanan Web Server,” *Jurnal Ilmiah NERO*, vol. 1 no. 3, pp. 190-197, 2015.