Jurnal Computech & Bisnis, Vol. 6, No. 2, Desember 2012, 60-68
ISSN 2442-4943

MANAJEMEN RESIKO PADA IMPLEMENTASI SAAS *(SOFTWARE AS A SERVICE)*

Toni Kusnandar
Magister Informatika – STEI,ITB Jl Ganesha 10 Bandung,
email:toni.kusnandar@gmail.com

Abstract

In recent years, Cloud Computing has evolved into a promising business concept as one of the segments of the industry's fastest growing of information technology. Many companies are increasingly aware that by leveraging the Cloud, they can gain quick access to the best business applications or drastically improve their infrastructure resources, with almost no additional cost. But as more and more individuals and corporate information stored in the Cloud, concerns began to grow about how safe the Cloud. Risk management to use SaaS selection decisions necessary so an organization can get the most of the advantages of cloud computing technologies. Risk analysis helps reduce uncertainty in decision making rational, and as long as there is uncertainty in decision making, risk evaluation will be an important component of the decision-making process. Reducing uncertainty can only be done by achieving and process more information on the initiative of the decision domain of risk analysis.

Keywords: *Cloud Computing, Software as a Service.*

Abstrak

Dalam beberapa tahun terakhir, cloud computing telah berkembang menjadi konsep bisnis yang menjanjikan sebagai salah satu segmen industri TIyang paling cepat berkembang. Banyak perusahaan semakin menyadari bahwa dengan memanfaatkan Cloud, mereka dapat memperoleh akses cepat ke aplikasi bisnis terbaik atau secara drastis meningkatkan sumber daya infrastruktur mereka, dengan hampir tanpa biaya tambahan. Tetapi karena semakin banyak informasi individu dan perusahaan disimpan di Cloud, kekhawatiran mulai tumbuh tentang seberapa aman Cloud itu. Manajemen resiko terhadap keputusan pemilihan pemanfaata SaaS perlu dilakukan agar sebuah organisasi dapat memanfaatkan semaksimal mungkin keunggulan teknologi Cloud Computing tersebut. Analisis risiko membantu mengurangi ketidakpastian dalam pengambilan keputusan rasional, dan selama ketidakpastian ada dalam pengambilan keputusan, risiko mengevaluasi akan menjadi komponen penting dari proses pengambilan keputusan.Mengurangi ketidakpastian hanya dapat dilakukan dengan mencapai dan mengolah informasi lebih lanjut terhadap domain keputusan dalam inisiatif analisis risiko.

Kata Kunci*: Cloud Computing, Software as a Service.*

1. PENDAHULUAN

Internet, sering digambarkan sebagai awan dalam diagram arsitektur, telah mengubah cara hidup bagi individu dan bisnis. Tulisan ini menyoroti masalah keamanan dalam *Cloud Computing* (komputasi awan), dengan fokus pada jaminan informasi, dan menyediakan strategi-strategi untuk mengadopsi ketika mengevaluasi penyedia layanan *cloud* dan ketika merancang, mengembangkan, dan menggunakan aplikasi yang akan beroperasi di *cloud*. Hal ini juga memberikan panduan tentang langkah-langkah berikutnya yang diperlukan untuk komputasi awan yang aman.

Karena paradigma komputasi awan masih berkembang, definisi umumnya tetap masih dalam progress. Definisi yang saat ini paling banyak diterima untuk komputasi awan adalah seperti yang dikembangkan oleh National Institute of Standards and Technology (NIST):

Cloud Computing a*dalah sebuah model yang memungkinkan kenyamanan, berdasarkan permintaan akses jaringan ke sumber daya komputasi yang telah dikonfigurasi bersama (misalnya, jaringan, server, penyimpanan, aplikasi, dan jasa) yang dapat dengan cepat ditetapkan dan dirilis dengan upaya manajemen atau interaksidengan penyedia layanansecara minimal*. (Paul, Mano., 2011)

2. MODEL CLOUD COMPUTING

Walaupun istilah *Cloud Computing* ini banyak digunakan, perlu dicatat bahwa *Model Cloud* tidak sama. Dengan demikian, bahwa pendekatan keamanan *Model Cloud* (Arora, Pankaj., 2010) pada suatu organisasi tidak akan berlaku umum. *Model Cloud* dapat dibagi ke dalam :

- *Software as a Service* (Saas),
- *Platform as aService*(PaaS) dan
- *Integrationas a Service* (IaaS).

Ketika sebuah organisasi sedang mempertimbangkan tentang keamanan *Cloud Computing,*mereka harus mempertimbangkan semua perbedaan dan persamaan antara ketiga segmen *Model Cloud*tersebut.

Gambar 1. Model *Cloud Computing*

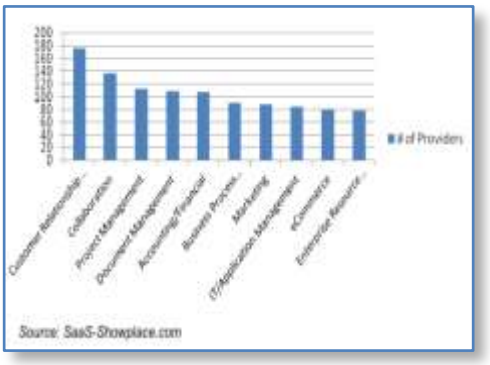
2.1 Software-As-A-Service

Software bukan sebagai layanan kadang-kadang disebut sebagai "*software on demand*" adalah perangkat lunak yang digunakan di internet / atau yang berjalan di belakang firewall pada jaringan lokal atau komputer pribadi. Dengan SaaS, penyedia lisensi aplikasi untuk pelanggan baik sebagai layanan sesuai permintaan, melalui berlangganan, dalam sebuah model "*pay asyou-go*". Pendekatan untuk pengiriman aplikasi adalah bagian dari utilitas model komputasi dimana semua teknologi dalam "*cloud*" ini diakses melalui Internet sebagai sebuah layanan. SaaS awalnya banyak digunakan untuk otomatisasi tenaga penjualan dan Customer Relationship Management (CRM). Sekarang telah menjadi biasa untuk tugas-tugas bisnis, termasuk penagihan terkomputerisasi, faktur, manajemen sumber daya manusia, keuangan, manajemen konten, kolaborasi, pengelolaan dokumen, dan manajemen *service desk*.

2.2 Platform-As-A-a Service

Platform as a Service (PaaS) adalah cara

untuk menyewa perangkat keras, sistem operasi, penyimpanan dan kapasitas jaringan melalui Internet. Model pelayanan memungkinkan pelanggan untuk menyewa server virtual dan layanan terkait untuk menjalankan aplikasi yang ada atau mengembangkan dan menguji yang baru. Platform as a Serrvice (PaaS) adalah hasil dari Software as a Service (SaaS), sebuah model distribusi perangkat lunak di mana aplikasi *host* perangkat lunak yang dibuat tersedia untuk pelanggan melalui Internet. PaaS memiliki beberapa keuntungan bagi pengembang. Dengan PaaS, fitur sistem operasi dapat seringdiubah dan ditingkatkan. Tim pengembangan yang didistribusikan secara geografis dapat bekerja sama dalam proyek pengembangan perangkat lunak. Layanan dapat diperoleh dari berbagai sumber yang melintasi batas internasional. Biaya awal dan berkelanjutan dapat dikurangi dengan penggunaan layanan infrastruktur dari satu vendor daripada mempertahankan fasilitas beberapa hardware yang sering melakukan fungsi ganda atau menderita karena masalah ketidakcocokan. Biaya keseluruhan juga dapat diminimalkan dengan penyatuan upaya pengembangan pemrograman. Pada sisi negatifnya, PaaS melibatkan beberapa risiko "*lock-in*" jika pelanggan memerlukan antarmuka layanan eksklusif atau bahasa pembangunan tertentu. Perangkap potensial lainnya adalah bahwa fleksibilitas penawaran mungkin tidak memenuhi kebutuhan beberapa pengguna yang persyaratannya cepat berkembang.



Gambar2. Top 10 SaaS provider categories, 2010.

2.3 Infrastruktur-as-a-Service

Infrastruktur as a Service adalah model ketentuan bahwa organisasi penyedia peralatan *outsourcing*digunakan untuk mendukung operasi, termasuk media penyimpanan, perangkat keras, server dan komponen jaringan. Service provider memiliki peralatan dan bertanggung jawab untuk melakukan *housing*, menjalankan dan merawatnya. Klien biasanya membayar berdasarkan pada apa yang dia gunakan saja (*pays on a per-use basis*).

Karakteristik dan komponen IaaS meliputi:

1. Layanan Utilitas komputasi dan model penagihan.
2. Otomatisasi tugas-tugas administratif.
3. Scaling secara Dinamis
4. Virtualisasi Desktop
5. Layanan berbasis Kebijakan.
6. Penghubung Internet

3. ISU KEAMANAN CLOUD COMPUTING

Dalam beberapa tahun terakhir, komputasi awan telah berkembang dari konsep bisnis yang menjanjikan sebagai salah satu segmen dari industri TIyang paling cepat berkembang. Saat ini, perusahaan yang terkena resesi semakin menyadari bahwa hanya dengan memanfaatkan cloud mereka dapat memperoleh akses cepat ke aplikasi bisnis atau meningkatkan sumber daya infrastruktur mereka secara drastis, dengan biaya yang sangat rendah. Tetapi karena semakin banyak informasi pada individu dan perusahaan ditempatkan di *cloud*, kekhawatiran mulai tumbuh terhadap seberapa aman lingkungan *cloud*itu.

3.1 Keamanan

Dimana data anda lebih aman, pada Hard Disk lokal atau pada server dengan tingkat keamanan yang tinggi di *cloud*?

Beberapa berpendapat bahwa data pelanggan lebih aman bila dikelola secara internal, sementara yang lain berpendapat bahwa penyedia *cloud* memiliki insentif yang kuat untuk menjaga kepercayaan dan dengan demikian menerapkan tingkat keamanan yang lebih tinggi. Namun, di *cloud*, data anda akan didistribusikan ke komputer-komputer individu terlepas dari mana repositori basisdata anda ini akhirnya disimpan. Hacker yang rajin dapat menyerang hampir server apapun, dan ada statistik yang menunjukkan bahwa sepertiga dari hasil pelanggaran dari laptop dan perangkat yang dicuri atau hilang serta dari data karyawan yang sengaja diekspos di Internet, hampir 16 persen akibat pencurian oleh internal.

3.2 Privacy

Berbeda dengan model komputasi tradisional, komputasi awan memanfaatkan teknologi komputasi virtual, data pribadi pengguna dapat tersebar di berbagai pusat data virtual daripada tinggal di lokasi fisik yang sama, bahkan di seluruh perbatasan nasional, saat ini, perlindungan data pribadi akan menghadapi kontroversi dari sistem hukum yang berbeda. Di sisi lain, pengguna dapat membocorkan informasi tersembunyi ketika mereka mengakses layanan komputasi awan. Penyerang dapat menganalisa tugas penting tergantung pada tugas komputasi yang diajukan oleh pengguna.

3.3 Reliability

Server di awan memiliki masalah yang sama seperti server kita sendiri. Server di awan juga mengalami downtime dan slowdowns, perbedaannya dalam model komputasi awan adalah pengguna memiliki ketergantungan lebih tinggi pada *Cloud Service Provider* (CSP). Ada perbedaan besar dalam model layanan CSP, sekali kita memilih CSP tertentu, kita mungkin akan terkunci (*lock-in*) didalamnya, sehingga membawa risiko potensial bisnis yang aman.

3.4 Masalah Hukum

Terlepas dari upaya untuk membawa ke dalam jalur yang sah, seperti tahun 2009, pemasok *Amazon Web Services* menyediakan layanannya dengan mengembangkan jalan terbatas dan *rail network*serta membiarkan penggunanya untuk memilih "*availability zones*" sendiri. Di sisi lain, kekhawatiran terhadap tahapan keamanan dan kerahasiaan dari individu sampai tingkat legislative tetap ada. (Ahmadi, A., 2009)

3.5 Open Standard

Open Standard (standar terbuka) sangat penting untuk pertumbuhan komputasi awan. Penyedia awan yang paling mengekspos API yang biasanya terdokumentasi dengan baik, tetapi pelaksanaannya juga unik sehingga tidak *interoperable*. Beberapa vendor telah mengadopsi API lain dan ada sejumlah standar terbuka sedang dikembangkan, termasuk *Open Cloud Computing Interface* (OGF). *Open Cloud Consortium* (OCC) bekerja untuk mengembangkan konsensus dan praktek standar komputasi awan.

3.6 Compliance

Banyak peraturan yang berkaitan dengan penyimpanan dan penggunaan data memerlukan pelaporan yang teratur dan kegiatan audit, penyedia awan harus memungkinkan pelanggan mereka untuk mematuhi peraturan tersebut dengan tepat. Mengelola Kepatuhan dan Keamanan untuk *Cloud Computing*, memberikan wawasan tentang bagaimana pandangan top-down dari semua sumber daya TI dalam lokasi berbasis *cloud* dapat memberikan manajemen yang kuat dan penegakan kebijakan kepatuhan. Selain persyaratan yang dikenakan pelanggan, pusat data yang dikelola oleh penyedia awan juga dapat dikenakan persyaratan kepatuhan.

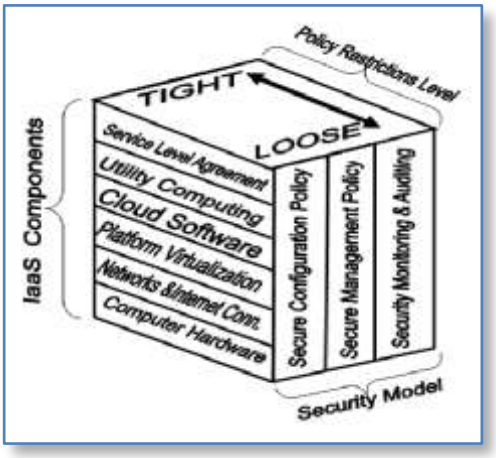
3.7 Kebebasan

Cloud computing tidak memungkinkan pengguna untuk secara fisik memiliki

penyimpan data, meninggalkan penyimpan data dan kontrol tetap berada di tangan penyedia *cloud*. Pelanggan akan berpendapat bahwa ini cukup mendasar dan memberi mereka kemampuan untuk menyimpan salinan data mereka sendiri dalam bentuk yang mempertahankan kebebasan pilihan mereka dan melindungi mereka terhadap isu-isu tertentu di luar kendali mereka,selain itu mewujudkan komputasi awan dapat membawa manfaat yang luar biasa.

3.8Long Term Viabilitas

Kita harus memastikan bahwa data yang kita masukkan ke dalam awan tidak akan pernah menjadi tidak valid ketika penyedia komputasi awan kita bangkrut atau bisa diperoleh dan diambil oleh perusahaan yang lebih besar. Tanyakan bagaimana kita akan mendapatkan data kita kembali dan jika memungkikan dalam format yang bisa kita impor ke dalam aplikasi pengganti.



Gambar 3. Security Model In IAAS

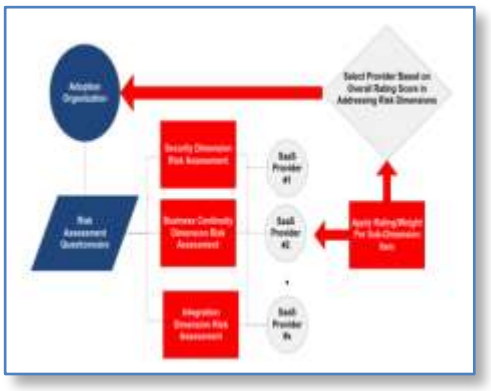
4. KEAMANAN UNTUK MODEL IAAS

Model Keamanan untuk IaaS (SMI) sebagai panduan untuk menilai dan meningkatkan keamanan di setiap lapisan model pengiriman IaaS seperti yang ditunjukkan pada Gambar 3. SMI model terdiri dari tiga sisi: IaaS komponen, model keamanan, dan tingkat pembatasan. Sisi model keamanan mencakup tiga entitas vertikal di mana setiap entitas meliputi seluruh komponenIaaS. Entitas pertama adalah Kebijakan Konfigurasi Keamanan (SCP) untuk menjamin konfigurasi yang aman untuk setiap lapisan IaaS dalam Hardware, Software, atau konfigurasi SLA, biasanya insiden kesalahan konfigurasi bisa membahayakan keamanan seluruh sistem. Yang kedua adalah Kebijakan Manajemen Sumber Daya Keamanan(SRMP) yang mengontrol peran manajemen dan hak istimewanya. Entitas terakhir adalah Kebijakan Monitoring Keamanan dan Pemeriksaan Keuangan (SPMA) yang signifikan untuk melacak siklus hidup sistem. Sisi kebijakan pembatasan menentukan tingkat pembatasan untuk pembatasan keamanan entitas model dimulai dari longgar ke ketat, ketergantungan pada operator, klien, dan kebutuhan pelayanan. Namun demikian, kita berharap model SMI menjadi awal yang baik untuk standardisasi lapisan IaaS. Model ini menunjukkan hubungan antaracomponents IaaSdan persyaratan keamanan, dan memudahkan peningkatan keamanan di lapisan individu untuk mencapai sistem keamanantotal IaaS.

5. SaaS Cloud Risk Assessment Framework

Penentuan relevansi awal untuk memvalidasi SaaS Cloud Risk Assessment (S-CRA) framework sesuai dengan teori yang ada mengenai pentingnya memastikan signifikansi risiko untuk fungsi organisasi sebelum diintegrasikan ke dalam analisis risiko dan framework manajemen risiko. Koller (2005) menyatakan bahwa mendefinisikan risiko yang relevan adalah langkah penting ke arah yang benar terhadap penilaian risiko yang efektif. Relevansi ini hanya ditentukan, seperti disebutkan dalam literatur teori keputusan, oleh pengamatan empiris dari hasil sebelumnya dan mirip dengan memvalidasi pengaruh mereka pada

hasil yang diharapkan dan yang diperkirakan. Dimensi risiko yang relevan dapat digunakan sebagai masukan ke dalam proses pengambilan SaaS dalam bentuk kuesioner detil tentang unsur risiko. Manajemen pengambil keputusan dapat menggunakan kuesioner jawaban dari masing-masing penyedia SaaS pada latihan evaluasi obyektif dengan menerapkan peringkat kepastian dan / atau bobot unsur probabilitas untuk memilih penyedia berdasarkan utilitas optimal, dinyatakan sebagai skor nilai keseluruhan untuk masing-masing operator. Konstruksi dari model S-CRA, didalamnya termasuk dimensi risiko dan sub dimensi, kepuasan perangkat lunak, kepastian risiko, risiko yang relevan, peringkat risiko, faktor bobot, dan skor rating risiko secara keseluruhan.



Gambar 4. SaaS Cloud Risk Assessment (S-CRA) framework

Gambar 4. menunjukkan *frameworkSaaS Cloud Risk Assessment* (S-CRA), yang membutuhkan penggunaan kuesioner penilaian risiko untuk mengevaluasi dan menilai setiap penyedia solusi SaaS potensial di bidang yang relevandengan risiko keamanan, kelangsungan bisnis, dan integrasi dalam rangka memperoleh skor risiko komposit untuk masing-masing operator untuk digunakan sebagai faktor seleksi diskriminatif.

5.1 Analytical Network Process and Analytical Hierarchy Process.

Proses analisis hirarki (AHP) dan proses analisis jaringan (ANP) keduanya merupakan*framework*pengambilan keputusan yang dikembangkan oleh Saaty (1980) untuk memecahkan masalah pengambilan keputusan yang kompleks yang melibatkan beberapa kriteria, objektif, dan tujuan. AHP dan ANP dibedakan oleh adanya ketergantungan antara kriteria yang digunakan dalam proses evaluasi. AHP bergantung pada struktur hirarki keputusan sederhana tanpa ketergantungan antar kriteria, sedangkan kerangka ANP lebih kompleks yang akan bergantung pada ketergantungan antara kriteria pengelompokan. Menerapkan *framework*keputusan AHP melibatkan penataan keputusan secara hirarki ke sublevels lebih kecil dan independen untuk memudahkan analisis dan pemahaman. Pada tingkat atas adalah pernyataan dari tujuan keputusan. Tingkat berikutnya menunjukkan kriteria dan prioritas untuk setiap kriteria dinyatakan sebagai faktor subjektif berat atau skor dari 1 sampai 9 dengan pembuat keputusan. Tingkat terakhir menunjukkan alternatif dan prioritas atau skor untuk setiap alternatif dan memungkinkan pembuat keputusan dengan mudah memilih alternatif terbaik berdasarkan prioritas secara keseluruhan yang dikombinasikan atau skor pada semua kriteria. ANP digunakan dalam skenario optimasi keputusanyang lebih kompleks, di mana ketergantungan ada pada berbagai kriteria, tetapi metodologi yang sama berlaku (Lee dan Kim, 2000). Sebagai hasil dari pengelompokan kriteria yang saling tergantung dalam format jaringan, memecahkan masalah keputusan menggunakan ANP jelas lebih kompleks dan memerlukan penggunaanrumusmatriks matematika yang kompleks untuk mendapatkan alternatif yang optimal.

AHP dan ANP adalah framework yang telah terbukti dan efektif untuk melakukan seleksi perangkat lunak dan teknologi pada umumnya, tetapi mereka memiliki kelemahan utama yang membuat mereka tidak praktis sebagai

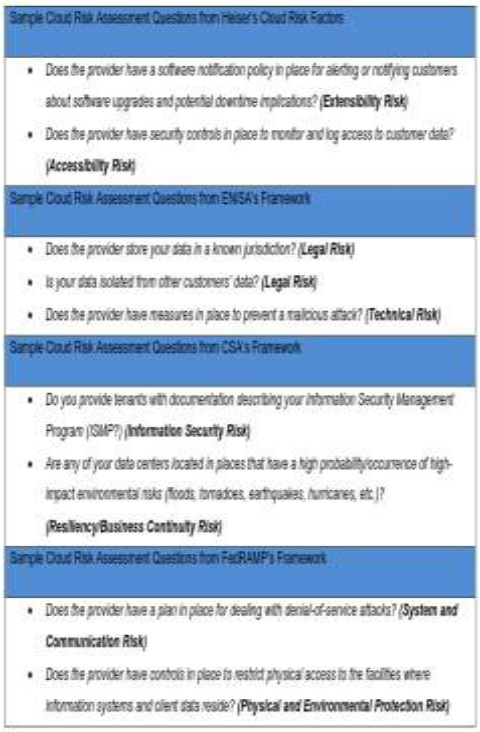
alat keputusan SaaS: Kedua framework membutuhkan pengambil keputusan untuk melihat setiap kriteria yang bersaing terhadap yang lain dan untuk membuat sebuah penilaian satu prioritas di atas yang lain. Namun, dalam pemilihan SaaS, dimensi risiko keamanan, kelangsungan bisnis, dan integrasi biasanya dianggap sama penting (Heiser, 2010a). Memaksakan pembuat keputusan untuk menentukan risiko lebih penting daripada yang lain mengurangi efektivitas proses evaluasi SaaS.

5.2. COTS Software Evaluation Methods.

Hollander R^{2}ISC ("*risk-squared*") adalah metodologi evaluasi menyeluruh yang didasarkan pada pendekatan siklus-hidup untuk pemilihan software COTS (2000). *Risk-squared* menyarankan menggunakan empat faktor untuk mengukur *adoptability* perangkat lunak :

- kemampuan perangkat lunak untuk memenuhi persyaratan yang ditetapkan,
- kemampuanuntuk disesuaikan dan mudah diimplementasikan;
- kemampuan vendor untuk mendukung perangkat lunak
- biaya total kepemilikan perangkat lunak.

Metodologi ini menyediakan daftar yang mendalam tentang subkriteria standar untuk setiap faktor. Proses *risk-squared* dimulai dengan menyusun persyaratan dalam *request for proposal* (RFP) dan mengundang vendor potensial untuk merespon RFP dengan proposal formal dalam format yang telah ditentukan oleh RFP. Metode *risk-squared* kemudian mengharuskan pengambil keputusan untuk menilai, dalam skala 10-point, kemampuan masing-masing vendor perangkat lunak untuk memenuhi kebutuhan masing-masing seperti yang dijelaskan dalam RFP dan memberikan peringkat prioritas untuk setiap kebutuhan berdasarkan kebutuhan bisnis. Seperti framework AHP dan ANP, *risksquared* menggunakan peringkat individu sebagai masukan ke formula yang berasal dari peringkat risiko akhir dan berbeda untuk setiap pilihan perangkat lunak.



Gambar 5. Contoh pertanyaan penilaian risiko berdasarkan faktor-faktor risiko awan Heiser dan kerangka ENISA, CSA, dan penilaian risiko awanFedRAMP.

Terlepas dari panduan rinci yang bertujuan mengurangi kesalahan dalam pemilihan perangkat lunak, *risk-squared* lebih cocok untuk pemilihan software COTS yang telah ada dan untuk organisasi yang dapat menanggung waktu dan biaya komitmen untuk melaksanakan pendekatan ini.Menetapkan pengambilan keputusan dibutuhkan, bersama dengan beberapa elemen panduan seleksi *risk-squared*, dapat diadopsi untuk kegiatan seleksi SaaS, tapi pendekatan tersebut tidak mempertimbangkan persyaratan keamanan atau keseluruhan risiko *outsourcing*perangkat lunak.Selain itu, paradigma akuisisi modern perangkat lunak, berbasis *cloud* terutama SaaS, yang melibatkan penyewaan perangkat

lunak dan penggunaannya melalui browser web, bukan membeli, memiliki, dan menggunakannya pada internal infrastruktur organisasi TI, tidak membutuhkan pendekatan evaluasi yang berkepanjangan, seperti yang disarankan oleh metode *risk-squared*.

Procurement Oriented Requirements Engineering (PORE) adalah metode seleksi lainnya yang berlaku untuk pemilihan perangkat lunak COTS, diusulkan dalam penelitian oleh Maiden dan Ncube (1998). Metode PORE diambil dari berbagai teknik pemilihan perangkat lunak konvensional dan pendekatan keputusan, termasuk teknik analisis fitur untuk mencetak kemampuan sistem yang dievaluasi sertauntuk memenuhi persyaratan kedua metode seleksi ANP dan AHP (Kitchenham, Pickard, Linkman, & Jones, 2005). PORE memperkenalkan tiga template yang berbeda untuk mengevaluasi sistem perangkat lunak dalam keadaan yang berbeda.

- Template 1 sangat membantu dalam menyusun persyaratan pelanggan awal dan penyaringan alternatif produk berdasarkan informasi yang diberikan oleh pemasok.
- Template 2 menyediakan bantuan yang sama seperti template 1 tapi menyarankan menggunakan kasus uji untuk kebutuhan individu berdasarkan demonstrasi pemasok.
- Template 3 adalah mirip dengan dua lainnya tetapi memberikan pedoman akuisisi berdasarkan informasi dari pelanggan dimulai dari penelitian produk.

Gambar 6. Disintesis dimensi dan sub-dimensi risiko SaaS

Nilai inti dari PORE adalah *requirements-centric approach*dan penggunaan tujuan kebutuhan untuk mengarahkan proses pemilihan dan menolak calon sistem.Tujuannya mencakup persyaratan yang penting, persyaratan yang tidak penting, kompleks, dan pengguna. PORE merupakan proses empat langkah, dimulai dengan memperoleh informasi tentang persyaratan dan produk yang saling melengkapi.Tiga langkah terakhir proses PORE mencerminkan proses konvensional seleksi perangkat lunak COTS, serta proses normatif keputusan, langkah ini mencakup menganalisis informasi tercapai, menggunakan teknik peringkat dan skor untuk mengevaluasi sesuai dengan persyaratan, dan menolak secara iteratif dan mempersempit daftar sistem compliant sampai pilihan ditetapkan.

6.KESIMPULAN

Konsensus keputusan dalam literatur teori adalah bahwa analisis risiko membantu mengurangi ketidakpastian dalam pengambilan keputusan rasional, dan selama ketidakpastian ada dalam pengambilan keputusan, risiko mengevaluasi akan menjadi komponen penting dari proses pengambilan keputusan. Hofstede, G (1998) mendefinisikan tentang persepsi ketidakpastian ancaman dan kecemasan karena tidak diketahui, faktor risiko tidak terstruktur, atau ambigu.

Dalam hal hubungan antara ketidakpastian dan risiko, ketidakpastian memperkenalkan unsur risiko ke dalam proses pengambilan keputusan tetapi tidak memberikan kontribusi pada dampak atau kemungkinan risiko. Untuk akuisisi SaaS, ketidakpastian seputar penerapan outsourcing teknologi perangkat lunak suatu organisasi dan dampak potensial yang mengganggu karena tindakan ini dapat memiliki pertimbangan yang memerlukan elemen risiko dalam proses akuisisi perangkat lunak.Mengurangi ketidakpastian hanya

dapat dilakukan dengan mencapai dan mengolah informasi lebih lanjut terhadap domain keputusan dalam inisiatif analisis risiko. Gilboa (2009) sependapat dalam menyatakan bahwa pengambilan keputusan rasional harus mengumpulkan informasi yang relevan untuk mengurangi ketidakpastian. Namun demikian, seperti dicatat oleh Holford (2009), bisnis sering bergeser ke pengambilan keputusan fungsional tanpa melihat lebih dekat tentang ambiguitas dari situasi pengambilan keputusan yang dapat menyebabkan konsekuensi dan peningkatan risiko.

REFERENSI

Ahmadi, A. (2009). Teknologi Jaringan Dalam Cloud Computing. *Jurnal Computech & Bisnis*, *3*(1), 14-22.

Arora, Pankaj. (2010). Cloud Computing Security Issues inInfrastructure as a Service. *International Journal of Advanced Research in Computer Science and Software Engineering.*

Bernard, Leonel. (2011). *A Risk Assessment Framework for Evaluating Software-as-a-Service (SaaS) Cloud Services BeforeAdoption*. University of Maryland University.

Gilboa, I. (2009). *Theory of decision under uncertainty* (Vol. 1). Cambridge: Cambridge university press.

Hofstede, G. (1998). Attitudes, values and organizational culture: Disentangling the concepts. *Organization studies*, *19*(3), 477-493.

Kitchenham, B. A., Pickard, L., Linkman, S., & Jones, P. (2005). A framework for evaluating a software bidding model. *Information and Software Technology*,*47*(11), 747-760.

Ncube, C., & Maiden, N. A. (1999, May). PORE: Procurement-oriented requirements engineering method for the component-based systems engineering development paradigm. In *International Workshop on Component-Based Software Engineering* (pp. 130-140).

Paul, Mano. (2010). CSSLP, CISSP, AMBCI, MCAD, MCSD, Network+, ECSA M., *Security in the Skies - Cloud Computing Security Concerns, Threats, and Controls*. ISC.

Saaty, T.L., (1980). The Analytic Hierarchy Process. McGraw-Hill, Inc.,