PERBANDINGAN KUALITAS KEAMANAN OSS DENGAN CSS MENGGUNAKAN PROTOTIPE MODEL KUALITAS KEAMANAN BERBASISKAN MODEL KOMPETISI DAN PENAMBALAN

Ruktin Handayani, Aditya Ideawan
Teknik Informatika, Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
Email :ruktin11@mhs.if.its.ac.id aditya11@mhs.if.its.ac.id

ABSTRAK

Kualitas sebuah perangkat lunak adalah faktor yang menjadi perhatian penting ketika sebuah perangkat lunak dikatakan layak untuk digunakan. Keamanan (security) merupakan salah satu faktor kualitas yang cukup signifikan menjadi bahan pertimbangan untuk menentukan jenis perangkat lunak yang akan digunakan. Adalah sebuah tantangan bagi perangkat lunak sumber terbuka (Open Source Software, disingkat OSS) untuk dapat berdiri sukses dalam komunitas dan bersaing dengan perangkat lunak sumber tertutup (Close Source Software, disingkat CSS). Adanya anggapan atau mitos yang menyoroti kelemahan OSS menjadikan alasan tersendiri untuk melakukan pembuktian bahwa kualitas yang dimiliki oleh OSS tidak lebih buruk dibandingankan CSS. Beberapa metode digunakan untuk membandingkan kualitas keamanan pada OSS dan CSS. Dalam penelitian ini diusulkan sebuah pemodelan rumus baru yang dikhususkan untuk menghitung kualitas keamanan sebuah perangkat lunak yaitu model Kompetisi dan Penambalan (Competition and Patching) serta rumus McCall sebagai pengujian. Kuisoner akan sangat membantu untuk memperoleh data yang konkrit dan objektif serta mempermudah perhitungan. Dengan demikian kuisoner akan menjadi kontribusi berarti dalam penelitian ini.

Kata Kunci : *Open Source Software* (OSS), *Close Source Software* (CSS), Keamanan (*Security*), Kompetisi dan Penambalan (*Competition and Patching*), McCall, Kuisioner.

1. PENDAHULUAN

1.1 Latar Belakang

Perangkat lunak sumber terbuka (*Open Source Software,* disingkat OSS) merupakan perangkat lunak yang tidak memerlukan biaya dan dapat dimodifikasi dan didistribusikan kepada siapapun. Aplikasi dan komponen open source sangat penting bagi komunitas pengembang perangkat lunak. Pengembangan aplikasi menggunakan komponen open source dapat menghemat faktor biaya dan waktu. Penggunaan komponen sumber terbuka ini juga memiliki kualitas yang tinggi karena komponen-komponennya diuji secara langsung oleh seluruh masyarakat sebagai pihak pengguna. Kode sumber (*source code*) yang disediakan oleh pengembang komponen atau aplikasi memungkinkan pengembang lain dengan cepat memodifikasi kode untuk dapat memecahkan permasalahan mereka sendiri.

Kualitas merupakan masalah penting bagi OSS. OSS harus sesuai dengan standar mutu untuk dapat sukses dalam komunitas. Model kualitas McCall adalah model referensi untuk kualitas perangkat lunak. Dalam jurnal *Quality Standards in Open Source Lifecycle* [1], model ini disajikan dan dijelaskan. Demikian pula dengan alternatif modelnya, juga disajikan dan dijelaskan. Perbedaan antara model kualitas ini juga disorot. Pengaruh dalam memenuhi standar kualitas juga dibahas. Terdapat banyak keuntungan mengembangkan OSS atau menggunakan modul sumber terbuka, salah satunya adalah hal keamanan (*security*).

Masalah keamanan merupakan masalah yang sangat kompleks dan rumit. Model pengembangan memiliki efek yang netral terhadap masalah keamanan. Setiap model pengembangan memiliki peranan masing-masing yang penting dalam menciptakan perangkat lunak yang handal dan aman dan setiap model memiliki kelebihan dan kekurangan masing-masing yang saling melengkapi satu sama lain.

Isu-isu keamanan yang dihadapi sistem sumber terbuka, mencakup beberapa filosofi keamanan umum dan bagaimana membuat sistem tersebut lebih aman dari para penyusup. Beberapa pengguna komputer yang merupakan anggota dari komunitas pengguna OSS berpendapat bahwa kode program mereka lebih aman karena kelemahan kode program mereka lebih mudah ditemukan dan

diperbaiki oleh pemakai program tersebut. Sementara itu, komunitas perangkat lunak sumber tertutup (Close Source Software, disingkat CSS) berpendapat bahwa pembukaan akses ke kode program pada OSS akan memudahkan bagi beberapa kelompok tertentu untuk menyerang program tersebut.

Adanya mitos bahwa CSS memiliki kualitas keamanan lebih baik, mendorong penulis untuk melakukan pengujian kualitas keamanan OSS yang akan dibandingkan dengan CSS dengan studi kasus web browser. Pengujian kualitas dilakukan dengan menggunakan rumus tertentu dengan data hasil kuisioner. Perbandingan kualitas ini dikhususkan untuk faktor keamanan.

1.2 Hipotesis

Berdasarkan pengamatan yang dilakukan, penulis mengambil sebuah hopotesis tentang kualitas keamanan terhadap OSS dan CSS. Hipotesis tersebut adalah **"OSS memiliki kualitas keamanan lebih baik daripada CSS".**

2. REFERENSI TERKAIT

2.1 Perbandingan Model Pengukuran Kualitas Keamanan

Model kualitas keamanan yang paling sering direferensi oleh peneliti maupun ahli adalah ISO 9126 dan model kualitas McCall. Kedua model ini mampu memberikan gambaran tentang kualitas perangkat lunak secara umum. Model kualitas yang khusus mengenai keamanan sampai saat ini masih diteliti. Menurut hasil pencarian kami, sudah banyak acuan-acuan model kualitas maupun pembuktian-pembuktian empiris tentang keamanan perangkat lunak.

Penelitian-penelitian [2][3] yang berhubungan dengan kualitas keamanan, kami menggunakan model yang digunakan dalam penelitian empiris kompetisi dan penambalan celah keamanan [2]. Penelitian tersebut kami pilih karena memiliki fitur yang tidak ditemukan pada model McCall.

Untuk mendapatkan hasil perbandingan model kualitas keamanan yang lain maka dapat dilihat kelebihan, kelemahan, kesempatan dan ancaman (SWOT) terhadap model kualitas yang dibandingkan.

2.1.1 Model McCall

Model ini merupakan model tertua dan paling sering digunakan dalam pengukuran kualitas perangkat lunak [4]. Model ini memiliki 11 faktor kualitas yang dibagi menjadi 3 kategori.

Kategori pertama, operasi produk dengan *correctness, reliability, integrity* dan *usability* sebagai faktor kualitas. Kategori kedua, revisi produk dengan *maintainability*, *flexibility* dan *testability* sebagai faktor kualitas. Kategori terakhir, transisi produk meliputi *portability*, *reusability* dan *interoperability*. Menurut penelitian [3] faktor kualitas keamanan dalam model ini adalah *correctness* dan *reliability*.

Model McCall merupakan model yang umum dipakai perbandingan karena kedua model tersebut sudah teruji, namun hasil pengukuran kualitas yang dihasilkan bersifat umum, tidak spesifik untuk menguji kualitas keamanannya. Kedua model dapat berkembang dengan membuat model baru yang berbasiskan salah satu untuk menguji kualitas keamanan.

2.1.2 Model Kompetisi dan Penambalan

Penelitian ini menggunakan model analisis empiris yang membandingkan jumlah pesaing langsung, pesaing tidak langsung, keterbukaan informasi terhadap celah dan besaran pasar terhadap lama waktu yang dibutuhkan pengembang untuk merilis perbaikan terhadap celah keamanan. Model ini mengansumsikan bahwa pengguna akhir mendapatkan jaminan kualitas keamanan dengan munculnya rilis terbaru atau patch perangkat lunak.

Variabel terikat adalah durasi. Durasi menyatakan jumlah hari yang digunakan vendor untuk melepas tambalan (*patch)*. Waktu mulai dari celah keamanan diketahui baik secara langsung maupun tidak langsung sampai tambalan secara resmi dirilis. Variabel bebas yaitu rival, non-rival, kuantitas dan keberbahayaan (*severity*).

Rival menyatakan jumlah produk pesaing, sedangkan non-rival menyatakan jumlah produk lain yang memiliki permasalahan keamanan serupa atau menggunakan basis program yang sama. Kualitas menyatakan jumlah penjualan dalam satuan waktu. Keberbahayaan menyatakan tingkat ancaman keamanan yang diakibatkan oleh celah keamanan.

Model ini dibuat untuk melihat keterkaitan kualitas keamanan yang bersifat konformasi yaitu dari keaktifan pengembang dalam memberikan perbaikan agar perangkat lunak tetap berjalan dengan benar dan baik.

Model kompetisi dan penambalan memiliki keunikan karena mengukur kualitas keamanan dari seberapa aktif pengembang memperbaiki perangkat lunak ketika perangkat lunak tersebut berjalan. Bagi pengguna akhir, layanan purna jual dapat dijadikan pertimbangan aspek kualitas. Model ini memiliki kelemahan yaitu tidak dapat mengukur kualitas berdasarkan sisi desainnya karena faktor yang diamati merupakan kualitas konformasi. Selain itu, model ini hanya mengamati jumlah tambalan yang dirilis sehingga tidak dapat mengamati apakah rilis tersebut benar-benar meningkatkan kualitas keamanan atau tidak.

2.1.3 Model Pendekatan *Top-Down*

Model pendekatan *top-down* merupakan turunan dari model Mc Call untuk menguji

keamanan, dimana ini merupakan kekuatan utama dari model ini. Hanya saja, model ini masih dalam tahap pengembangan dan beberapa kriteria atribut non-fungsionalnya memiliki subkriteria yang direferensi oleh kriteria lainnya sehingga dapat menimbulkan bias atau kesulitan untuk perhitungan kualitasnya. Model ini memiliki kesempatan untuk direvisi sehingga menutupi semua kelemahannya, yang juga dapat dilihat bahwa model ini dapat ditinggalkan karena tidak ada yang mau mengembangkan.

Tabel 1 Perbandingan Model Kualitas Keamanan

Fitur	Model Kualitas Keamanan	
	McCall	Kompetisi & Penambalan
correctness	√	
reliability	√	
prediksi perbaikan		√

Tabel 1 menunjukkan perbandingan model kualitas keamanan dengan menggunakan rumus perhitungan McCall dan dan Penambalan. McCall mencakup faktor correctness dan reliability sedangkan model dan Penambalan akan dikenai prediksi perbaikan.

2.2 Vulnerabilities dalam Web Browsers

Vulnerabilities pada web browser adalah kelemahan atau cacat rancangan dari program dimana dapat dimanfaatkan oleh penyerang untuk melemahkan performa sistem atau untuk mendapatkan akses secara ilegal dengan cara *exploiting* [5]. Klasifikasi *vulnerability* pada web adalah *Cross-site Scripting* (XSS), *Denial-of-Service* (DOS), *Buffer Overflow*, *Remote Code Execution* [5]. Menurut laporan terkini [6], jumlah serangan meningkat dari tahun sebelumnya, dan diprediksi tren tersebut mengalami peningkatan untuk tahun-tahun yang akan datang.

2.3 OSS Vs CSS

Komunitas pendukung OSS mengatakan bahwa model pengembangan dengan OSS merupakan hal yang baik untuk meningkatkan kehandalan dan keamanan suatu perangkat lunak karena akan banyak pihak yang terlibat dalam perbaikan kesalahan-kesalahan serta kelemahan-kelemahan perangkat lunak tersebut. Pendapat ini memperoleh tentangan dari pihak-pihak lain, terutama pihak pengembang CSS karena terbukanya akses ke kode program akan memberikan hal yang menguntungkan bagi para peretas (*hacker*) untuk dengan mudah mengetahui kelemahan-kelemahan suatu sistem dan menggunakan kelemahan-kelemahan tersebut untuk melakukan tindakan-tindakan yang dapat merugikan pengguna dan pembuat perangkat lunak. Jadi manakah yang lebih baik dalam hal keamanan, OSS atau CSS?.

Penelitian maupun laporan teknis telah membahas kelebihan dan kekurangan baik dari segi keamanan maupun segi lainnya, namun sebuah penelitian empisis [7] menunjukkan bahwa dari 6 kasus, hanya 3 yang memiliki rata-rata waktu untuk *disclosure* (informasi tentang sebuah kesalahan untuk diketahui).

2.4 Kuisioner

Kuesioner adalah pertanyaan terstruktur yang diisi sendiri oleh responden atau diisi oleh pewawancara yang membacakan pertanyaan dan kemudian mencatat jawaban yang berikan.

Pertanyaan yang akan diberikan pada kuesioner ini adalah pertanyaan menyangkut fakta dan pendapat responden, sedangkan kuesioner yang digunakan pada penelitian ini adalah kuesioner tertutup, dimana responden diminta menjawab pertanyaan dan menjawab dengan memilih dari sejumlah alternatif. Keuntungan bentuk tertutup ialah mudah diselesaikan, mudah dianalisis, dan mampu memberikan jangkauan jawaban.

Ciri-ciri kuisioner yang baik adalah:

- ada petunjuk jelas mengenai maksud diberikannya kuesioner;
- ada petunjuk jelas mengenai cara pengisian kuesioner;
- menggunakan kalimat yang mudah dimengerti dan tidak bias arti;
- menghindari pertanyaan yang tidak jelas, tidak perlu dan tidak relevan;
- menghindari pertanyaan yang sugestif, bernada menekan/mengancam, dan sebagainya;
- menggunakan urutan pertanyaan yang logis dan sistematis; dan
- merahasiakan identitas responden agar responden obyektif dalam menjawab.

2.5 Sampel dan Variabel Penelitian

Sampel adalah bagian dari sebuah populasi yang dianggap dapat mewakili dari populasi tersebut. Untuk menentukan besarnya sampel sebagai berikut:

1. apabila subjek kurang dari 100, lebih baik diambil semua sehingga penelitiannya penelitian populasi dan
2. apabila subjeknya lebih besar dapat diambil antara 10-15 % atau 20-25 %.

Variabel adalah objek penelitian, atau apa yang menjadi titik perhatian suatu penelitian.

2.6 Validitas dan Releabilitas

Kecermatan pengukuran sangat diperlukan untuk memenuhi kriteria sebuah penelitian yang dianggap sebagai penelitian ilmiah. Ada dua syarat utama yang harus dipenuhi oleh alat ukur untuk memperoleh suatu pengukuran yang cermat, yaitu validitas dan releabilitas.

Validitas artinya alat ukur yang digunakan dalam pengukuran, dapat digunakan untuk

mengukur apa yang hendak diukur. Uji validitas dimaksudkan untuk menguji ketepatan item-item dalam kuesioner, apakah item-item yang ada mampu menggambarkan dan menjelaskan variabel yang diteliti. Jadi validitas adalah seberapa jauh alat dapat mengukur hal atau subjek yang ingin diukur. Validitas diusahakan dengan pikiran logis, meminta pendapat orang yang ahli, menggunakan kelompok yang telah diketahui sifatnya, kriteria independen. Item yang digunakan dalam penelitian ini untuk selanjutnya diuji reliabilitasnya.

Reliabilitas artinya memiliki sifat dapat dipercaya, yaitu apabila alat ukur digunakan berkali-kali oleh peneliti yang sama atau oleh peneliti lain tetap memberikan hasil yang sama. Jadi reliabilitas adalah seberapa jauh konsistensi alat ukur untuk dapat memberikan hasil yang sama dalam mengukur hal dan subjek yang sama.

Reliabilitas mengandung 3 makna yaitu:

1. tidak berubah-ubah;
2. konsisten; dan
3. dapat diandalkan.

Reliabilitas diuji dengan cara:

1. tes-retes;
2. dua bentuk skala yang ekuvalen; dan
3. bagi-dua atau *split-half*.

2.7 Cara Pengolahan dan Analisis Data

Pengolahan data adalah suatu proses dalam memperoleh data ringkasan atau angka ringkasan dengan menggunakan cara-cara atau rumus-rumus tertentu. Pengolahan data bertujuan mengubah data mentah dari hasil pengukuran menjadi data yang lebih halus sehingga memberikan arah untuk pengkajian lebih lanjut.

Teknik pengolahan data dalam penelitian ini menggunakan formula sederhana dalam aplikasi excel dan didukung dengan tampilan grafik yang mempermudah pembcaan data.

Kegiatan pengolahan data meliputi:

1. Penyutingan (*Editing*)

Penyutingan adalah pengecekan atau pengoreksian data yang telah terkumpul, tujuannya untuk menghilangkan kesalahan-kesalahan yang terdapat pada pencatatan dilapangan dan bersifat koreksi.

2. Pengkodean (*Coding*)

Pengkodean adalah pemberian kode-kode pada tiap-tiap data yang termasuk dalam katagori yang sama. Kode adalah isyarat yang dibuat dalam bentuk angka atau huruf yang memberikan petunjuk atau identitas pada suatu informasi atau data yang akan dianalisis.

3. Pemberian skor atau nilai

Dalam pemberian skor digunakan skala yang sudah ditentukan sendiri oleh peneliti yang merupakan salah satu cara untuk menentukan skor.

4. Tabulasi

Tabulasi adalah pembuatan tabel-tabel yang berisi data yang telah diberi kode sesuai dengan analisis yang dibutuhkan. Dalam melakukan tabulasi diperlukan ketelitian agar tidak terjadi kesalahan. Tabel hasil Tabulasi dapat berbentuk:

a. **Tabel pemindahan**, yaitu tabel tempat memindahkan kode-kode dari kuesioner atau pencatatan pengamatan. Tabel ini berfungsi sebagai arsip.
b. **Tabel biasa**, adalah tabel yang disusun berdasar sifat responden tertentu dan tujuan tertentu.
c. **Tabel analisis**, tabel yang memuat suatu jenis informasi yang telah.

Analisis adalah memperkirakan atau dengan menentukan besarnya pengaruh secara kuantitatif dari suatu (beberapa) kejadian terhadap suatu (beberapa) kejadian lainnya, serta memperkirakan atau meramalkan kejadian lainnya. Kejadian dapat dinyatakan sebagai perubahan nilai variabel. Proses analisis data dimulai dengan menelaah seluruh data yang diperoleh melalui hasil kuesioner.

3. PROTOTIPE MODEL KUALITAS KEAMANAN

Model yang diajukan untuk menilai kualitas keamanan dalam penelitian ini adalah berupa prototipe dengan basis kompetisi dan penambalan keamanan. Dari model yang diajukan, dapat dikemukakan variabel-variabel sebagai berikut :

3.1 Rivals

Rivals (pesaing) adalah jumlah vendor aplikasi pesaing yang sejenis fungsi dan penggunanya. Vendor saingan dapat menggunakan basis program yang sama maupun tidak, contoh Apple Safari™ dengan Google Chrome™ menggunakan web engine yang sama dan Mozilla Firefox™ menggunakan web engine yang berbeda namun memiliki fungsionalitas yang sama baik dengan Safari maupun Chrome.

3.2 Nonrivals

Nonrivals (bukan pesaing) adalah jumlah vendor pesaing tidak langsung dimana aplikasi yang fungsionalitas dan/atau penggunanya berbeda namum menggunakan basis program yang sama dengan program, seperti komponen pendukung atau pustaka perangkat lunak yang sama. Contoh aplikasi yang bisa disebut pesaing tidak langsung seperti aplikasi Songbird™ yang menggunakan web engine yang sama dengan Firefox.

3.3 Disclosure

Disclosure (keterbukaan) adalah munculnya informasi tentang *vulnerability* (celah). *Disclosure* memiliki nilai dari tidak ada (*non-exist*), parsial (sebagian/informasi tertutup) dan penuh (diumumkan ke publik). Pada penelitian [2]

disclosure didefinisikan sebagai efek bertambahnya kemunculan informasi *vulnerability* yang dialami sebagian besar vendor aplikasi, sehingga hanya vendor yang terkena masalah yang akan membuat tambalannya.

3.4 Quantity

Quantity (Jumlah) dapat dinyatakan sebagai jumlah aplikasi yang beredar dalam pasar atau yang dipakai oleh penggunanya. Pada penelitian [2], jumlah didapatkan dari data sensus Amerika Serikat, namun masih dapat dicari alternatif seperti jumlah unduh resmi (*official download*) maupun dengan survei.

4. METODOLOGI

Dalam tulisan ini digunakan sebuah metode perhitungan kualitas keamanan dengan menggunakan rumus yang sudah ditentukan berdasarkan kriteria-kriteria yang dianggap memenuhi. Data untuk input kriteria diperoleh berdasarkan survey dan kuisioner dari responden yang dianggap memenuhi. Selanjutnya, untuk pengujian, akan dilakukan perbandingan dengan perhitungan rumus McCall. Berikut ini adalah rincian dari metodologi yang dilakukan.

4.1 Penentuan Kriteria dan Variabel Perhitungan

4.1.1 Model Kompetisi dan Penambalan

Penentuan kriteria dilakukan dengan cara mengikuti model prototipe kualitas keamanan berbasis kompetisi dan penambalan sesuai dengan uraian yang tersebut pada bab III, yaitu *rivals, non rivals, disclosure*, dan *quantity*.

Model perhitungan adalah model yang kami ajukan dengan rumuskan sebagai berikut.

$$F_i = w_1 D_i + w_2 R_i + w_3 NR_i + w_4 Q_i \quad (1)$$

dimana:

- w_1 sampai w_4 adalah bobot untuk masing-masing faktor. Nilai **w1** sampai **w4** jika dijumlah sama dengan 1;
- D_i adalah variabel nilai *Disclosure.* Dengan rentang nilai misalnya 1 sampai dengan 30;
- R_i adalah variabel *Rivals* . Dengan Rentang nilai misalnya 1 sampai dengan 40;
- NR_i**N** adalah variabel *Nonrivals.* . Rentang nilai 1 sampai dengan 40.
- Q_i adalah variabel *Quantity.* . Rentang nilai 1 sampai dengan 40.

Tahapannya adalah sebagai berikut:

tahap 1:tentukan bobot (w) dari setiap variabel (biasanya 0 <= w <= 1);

tahap 2:tentukan skala dari nilai setiap variabel (misalnya, 0 <= nilai variabel <= 40);

tahap 3:berikan nilai pada setiap variabel;

tahap 4:hitung nilai kualitas total dengan rumus.

Justifikasi nilai pembobotan kami lakukan berdasarkan tingkat kepentingan keempat variabel terhadap keamanan sebuah web browser berdasarkan landasan teori terkait yang telah diuraikan pada bab III. Skala yang diberikan sebesar 0-40 didasarkan pada jumlah terbesar jawaban dari responden yaitu 38 angka, sehingga kami lakukan pembulatan menjadi 40. Nilai pada setiap variabel diperoleh dari data hasil analisi jawaban kuisioner responden.

Penggunaan rumus penjumalan variabel disebabkan karena berdasarkan penelitian empiris [1], penambahan jumlah *rivals* dan *nonrivals* mengurangi jumlah durasi sekitar 8-10 hari. Jika ditambahkan dengan meningkatnya *disclosure* dapat berkurang selama 7 hari. Dan bertambahnya quantity sebanyak 10% dapat mengurangi durasi selama 3 hari. Dengan demikian, dengan menjumlahkan keempat variabel *rivals, nonrivals, disclosure*, dan *quantity* diasumsikan dapat mengurangi lebih banyak jumlah durasi sehingga waktu rilis perbaikan terbaru sebagai perbaikan celah semakin cepat, sehingga kualitas keamanan akan semakin meningkat.

Tabel 2 adalah hasil perhitungan dengan model rumus kompetisi dan penambalan untuk masing-masing web browser.

Tabel 2 Penilaian Keamanan Web Browser dengan Model Kompetisi dan Penambalan

Faktor	Internet Explorer		Firefox		Chrome		Safari		Opera	
	Bobot	Nilai (n)	Bobot	Nilai (n)	Bobot	Nilai (n)	Bobot	Nilai (n)	Bobot	Nilai (n)
Rival	3	18	3	22	3	23	3	14	2	23
Non Rival	1	22	1	20	1	19	1	20	1	19
Disclosure	4	2	4	24	4	31	4	26	4	17
Quantity	2	10	2	48	2	37	2	2	3	3
Hasil	104		278		286		170		142	

Model ini masih harus diuji kembali keakuratannya dan penentuan bobot untuk setiap variabel sehingga mungkin akan berbeda untuk setiap studi kasus.

4.1.2 Perhitungan Dengan McCall

Pendekatan *engineering* menginginkan bahwa kualitas perangkat lunak ini dapat diukur secara kuantitatif, dalam bentuk angka-angka yang mudah dipahami oleh manusia. Untuk itu perlu ditentukan parameter atau atribut pengukuran. Menurut taksonomi McCall, atribut tersusun secara hirarkis, dimana level atas (*high-level attribute*) disebut faktor (*factor*), dan level bawah (*low-level attribute*) disebut dengan kriteria (*criteria*).

Faktor menunjukkan atribut kualitas produk dilihat dari sudut pandang pengguna. Sedangkan kriteria adalah parameter kualitas produk dilihat dari sudut pandang perangkat lunaknya sendiri. Faktor dan kriteria ini memiliki hubungan sebab akibat (*cause-effect*). Rumus McCall dituliskan sebagai berikut:

$$F_a = w_1 c_1 + w_2 c_2 + \cdots + w_n c_n \quad (2)$$

dimana:

- **Fa** adalah nilai total dari faktor a;
- **wi** adalah bobot untuk kriteria i;
- **ci** adalah nilai untuk kriteria i.

Sedangkan tahapannya adalah sebagai berikut:

tahap 1: tentukan kriteria yang digunakan untuk mengukur suatu faktor;
tahap 2: tentukan bobot (w) dari setiap kriteria (biasanya 0 <= w <= 1);
tahap 3: tentukan skala dari nilai kriteria (misalnya, 0 <= nilai kriteria <= 20);
tahap 4: berikan nilai pada tiap kriteria;
tahap 5: hitung nilai total dengan rumus.
tahap 6: menjumlahkan seluruh hasil perhitungan setiap faktor, dengan rumus berikut:

$$F_t = Fa_1 + Fa_2 + \cdots + Fa_n \quad (3)$$

Tabel 3 Penilaian Keamanan Web Browser dengan Model *McCall*

Faktor	Bobot	Kriteria	Internet Explorer			Firefox			Chrome			Safari			Opera		
			Bobot	Nilai	Fa	Bobot	Nilai	Fa	Bobot	Nilai	Fa	Bobot	Nilai	Fa	Bobot	Nilai	Fa
Correctness	6	*Accuracy*	0,3	14	15	0,3	22	22	0,3	23	24	0,3	20	20	0,3	21	20
		Completeness	0,1	12		0,1	23		0,1	23		0,1	21		0,1	21	
		Up-to-dateness	0,2	13		0,2	19		0,2	23		0,2	22		0,2	23	
		Availability (respon time)	0,3	15		0,3	22		0,3	22		0,3	20		0,3	21	
		Coding&documentation guideliness	0,1	18		0,1	28		0,1	31		0,1	14		0,1	9	
		Compliance (consistency)	0,1	18		0,1	30		0,1	30		0,1	13		0,1	9	
Reliability	4	*System reliability*	0,6	18	13	0,6	24	23	0,6	25	23	0,6	17	20	0,6	16	21
		Application reliability	0,2	3		0,2	18		0,2	18		0,2	29		0,2	32	
		Computational failure recovery	0,2	7		0,2	24		0,2	24		0,2	22		0,2	23	
Hasil (Ft)			138,8			225,9			236,7			198,6			200		

Faktor-faktor untuk perhitungan rumus McCall sama dengan faktor-faktor rumus untuk prototipe model yang disusulkan yakni hanya faktor-faktor yang berhubungan dengan kualitas keamanan. Faktor-faktor tersebut adalah termasuk ke dalam kategori *correctness* dan *reliability*. Keempat faktor tersebut akan diturunkan menjadi kriteria-kriteria untuk selanjutnya dihitung masing-masing bobotnya, sebagaimana tampak pada tabel 3.

4.2 Studi Kasus

Yang mendasari alasan studi kasus pada web browser adalah sebagai berikut:

1. perkembangan cepat;
2. penyebaran pengguna merata;
3. teknologi dan kegunaan sejenis tetapi dengan basis yang berbeda; dan
4. *vurnerability* yang dihadapi sama walau berbeda *source*.

4.3 Data Set

Responden yang dilibatkan dalam pengambilan data kuisioner adalah mereka yang memiliki kualifikasi yang sama dan dianggap mampu memberikan kontribusi terhadap pertanyaan-pertanyaan yang ditujukan untuk menjawab permasalahan. Hal ini dirasa karena pemakai saja tidak cukup untuk berkontribusi dalam pemenuhan jawaban kuisioner yang dibutuhkan. Responden yang berhasil kami himpun dalam survei ini adalah sebanyak 69 dengan rincian sebagai berikut:

1. pengembang: @20 orang;
2. kontributor: @15 orang;
3. seller: @15 orang; dan
4. pengguna biasa: @19 orang.

4.4 Teknik Penyebatan Kuisioner

Penyajian penyebaran kuisioner dalam penelitian ini menggunakan fasilitas yang disediakan oleh Google Docs™ dimana responden dapat mengisi kuisioner secara online, dimana saja dan kapan saja. Hasil jawaban responden akan terangkum dalam bentuk data list yang sudah terstruktur sehingga memudahkan dalam analisis data.

5. HASIL DAN UJI COBA

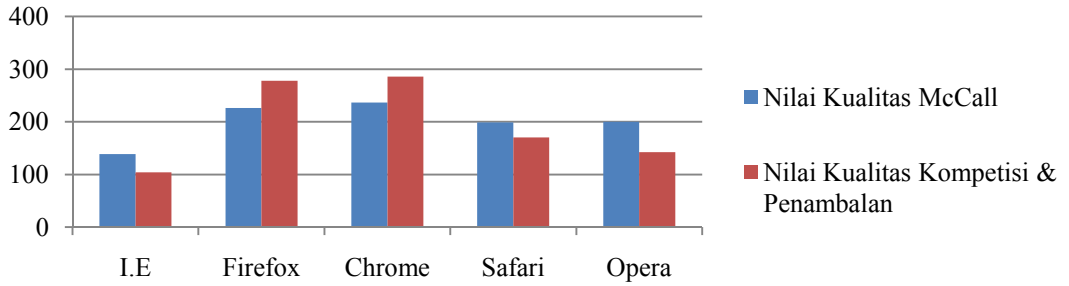
Setelah dilakukan analisis data kuisioner yang selanjutnya dijadikan sebagai nilai input untuk perhitungan model yang diusulkan dan perhitungan McCall maka uji coba dilakukan untuk membandingkan seberapa akurat hasil perhitungan dari pemodelan yang diusulkan dengan perhitungan McCall yang sudah menjadi standar pengukuran kualitas.

Tabel 4 Perbandingan Hasil Perhitungan Kualitas OSS vs CSS

No	Web Browser (OSS Vs CSS)	Nilai Kualitas	
		McCall	Kompetisi & Penambalan
1	I.E	138,8	104
2	Firefox	225,9	278
3	Chrome	236,7	286
4	Safari	198,6	170
5	Opera	200	142

Dari tabel 4 dapat dilihat perbandingan hasil perhitungan model rumus Kompetisi dan Penambalan dengan rumus McCall sebagai pembandingnya. Hasil perhitungan tersebut menunjukkan bahwa Chrome merupakan web browser berbasis OSS memiliki nilai kualitas keamanan tertinggi, yang selanjutnya diikuti oleh Firefox yang juga merupakan web browser berbasis OSS. Hal ini menunjukkan pula bahwa model rumus kompetisi dan penambalan patut diperhitungkan untuk digunakan dalam perhitungan kualitas sebuah perangkat lunak, khususnya kualitas keamanan.

Untuk lebih jelasnya, perbandingan hasil perhitungan model rumus Kompetisi dan Penambalan denan rumus McCall dapat dilihat pada gambar 1. Dari gambar 1 tampak bahwa Chrome dan Firefox menduduki batang tertinggi.



Gambar 1 Perbandingan Hasil Perhitungan Kualitas OSS vs CSS

6. ANCAMAN KEABSAHAN

Dari hasil penelitian yang diperoleh, diuji coba, maka selanjutnya dianalisis ancaman keabsahan yang dimungkinkan muncul dari model yang dihasilkan dalam penelitian ini. Beberapa kemungkinan yang dapat menyebabkan hasil perhitungan dari model rumus kompetisi dan penambalan menjadi tidak valid antara lain sebagai berikut.

Hasil yang diperoleh dalam penelitian ini adalah hasil dengan kondisi (kategori responden, jumlah responden, pengetahuan responden, jawaban responden) saat paper ini dibuat. Oleh sebab itu sangat mungkin sekali jika terjadi perubahan kondisi, maka akan menyebabkan hasil yang berbeda pula. Semakin banyak jumlah responden, jenis responden, maka semakin bagus hasil perhitungan yang dihasilkan.

Tingkat validasi pertanyaan kuisioner. Pertanyaan dalam kuisioner akan sangat mempengaruhi hasil jawaban kuisioner. Semakin valid pertanyaan kuisioner, maka akan semakin valid hasil dari perhitungan yang dihasilkan.

Tingkat validasi jawaban kuisioner. Jawaban kuisioner akan sangat mempengaruhi kualitas perhitungan yang dihasilkan. Jadi semakin valid jawabannya, maka semakin valid perhitungan kualitasnya.

Dari kemungkinan-kemungkinan tersebut di atas, maka diberikan antisipasi-antisipasi untuk mencegah hal tersebut di atas sebagai solusi, antara lain sebagai berikut.

1. Jumlah responden tidak boleh kurang dari jumlah responden dalam penelitian ini.
2. Perlu proses validasi pertanyaan kuisioner lebih lanjut.
3. Perlu proses validasi jawaban kuisioner lebih lanjut.

Perlu pengambilan data kuantitatif dan konkrit dari lembaga resmi yang menyediakan data-data mengenai jumlah variabel *rivals, nonrivals, disclosure, dan quantity* dari masing-masing web browser.

7. DISKUSI DAN PENELITIAN SELANJUTNYA

Perhitungan ini menunjukkan bahwa prototipe model rumus yang diusulkan dalam penelitian ini patut dijadikan pertimbangan untuk dimasukkan ke dalam model perhitungan kualitas khususnya keamanan. Hal ini disebabkan karena hipetesis yang dikemukakan dalam penelitian ini yakni OSS memiliki kualitas keamanan lebih baik dibandingkan CSS, dapat dibuktikan dengan perhitungan model rumus kompetisi dan penambalan berdasarkan pengujian dengan membandingkan dengan perhitungan rumus mcCall.

Pekerjaan selanjutnya adalah menjalankan survei, mengolah hasil dan meneliti dan memvalidasi lebih lanjut tingkat keakuratan dari model yang sudah diusulkan dalam penelitian ini. Kontribusi kuisioner setidaknya didukung oleh data dari lembaga penyedia data-data real dan akurat terkait nilai-nilai variabel yang dijadikan unsur dalam perhitungan rumus ini. Dengan demikian, kualitas dan keakuratan kuisioner dapat dinilai lebih objektif hasilnya.

8. DAFTAR PUSTAKA

[1] Vintila, B. (2010). "Quality Standards in Open Source Lifecycle". **Open Source Science Journal,** 46-55.

[2] Arora, A., Forman, C., Nandkumar, A., & Telang, R. (2010). "Competition and Patching of Security Vulnerabilities an Empirical Analysis". **Information Economics and Policy 22** , 164-177.

[3] Copigneaux, F., Verilog, T., & Martin, S. (1988). "Software Security Evaluation Based on a Top-Down McCall-Like Approach".

Aerospace Computer Security Applications Conference, 414-418.

[4] Fitzpatrick, R. (1996). "Software Quality: Definitions and Strategic Issues". 586-591. Dublin: Dublin Institute of Technology.

[5] Dhruwajita, D., Dhrubajyoti, P., & Sukumar, N. (2009). "Vulnerability in Web Browsers", Guwahati: Indian Institute of Techology

[6] Wood, P. (2011). **Symantec Internet Security Threat Report: 2011 Trends**. Symantec Inc.

[7] Schryen, G. (2009). "Security of Open Source and Closed Source Software: An Empirical Comparison of Publised Vulnerabilities". **15th Americas Conference on Information Systems**. 1-12.