الجمهورية العربية السورية
جامعة دمشق
كلية الهندسة الميكانيكية والكهربائية
قسم هندسة الحواسيب والأتمتة

تحسين موازنة الحمل في سحابة التخزين الخاصة

Enhancing Load Balancing in Private Cloud Storage

أعد هذا البحث لنيل درجة الماجستير في هندسة الحواسيب وشبكاتها

إعداد:

المهندس نزيه أحمد حرفوش

إشراف:

الدكتور أغيد القطعان

جامعة دمشق 2016م

الملخص

الحوسبة السحابية (Cloud Computing) تقنية تتيح نقل المعالجة ومساحة التخزين الخاصة بالحاسوب إلى ما يسمى السحابة وبهذا تتحول برامج تكنولوجيا المعلومات من منتجات دائماً ما تحتاج للصيانة والتطوير إلى خدمات تقدمها السحابة عبر الشبكة، وبذلك تساهم هذه التقنية في إبعاد مشاكل صيانة الموارد الفيزيائية والأنظمة وتطوير البرامج عن الشركات المُستخدمة لها، وبالتالي يتركز مجهود الجهات المستفيدة على استخدام هذه الخدمات فقط. تقدّم السحابة الخدمات للمستخدمين من خلال نشر الآلات الافتراضية في بنيتها التحتية لذلك فإن جودة الخدمات المُقدمة من السحابة تعتمد بالدرجة الأولى على طوبولوجيا نشر الآلات الافتراضية (Virtual Machines [VMs]) كما تعتمد على البنية التحتية للسحابة.

يهدف هذا البحث إلى تحسين جودة وكفاءة أداء الحوسبة السحابية من خلال تحسين طوبولوجيا نشر الآلات الافتراضية، والتي بدورها تلعب دوراً مهماً في تحقيق حمل متوازن ومستقر في السحابة لذلك تم دراسة وتحليل تقنيات وآليات نشر الآلات الافتراضية في الحوسبة السحابية وتم اقتراح وتطبيق آلية جديدة لنشر الآلات الافتراضية حيث قمنا بإضافة متحكم جديد (تطبيق برمجي) إلى بنية السحابة مهمته مراقبة حالة الشبكة (بين متحكمات السحابة والعقد من جهة وبين متحكمات السحابة والمستخدمين من جهة أخرى) بشكل مستمر ليقوم بتحديد الطوبولوجيا الأفضل لنشر الآلات الافتراضية.

تم بناء سحابة تخزين خاصة في المصرف التجاري السوري وذلك باستخدام المنصة (Eucalyptus) المفتوحة المصدر والموارد الفيزيائية المتاحة في المصرف التجاري السوري وتم تطبيق الآلية المقترحة وإجراء الاختبارات لتقييم الأداء، ثم قمنا بتحليل أداء الآلية وفق عدة حالات دراسية وأظهرت النتائج تحسناً ملحوظاً في الأداء لكل الحالات الدراسية كما أكدت النتائج أن الآلية المطبقة رفعت من كفاءة استثمار الموارد الفيزيائية للسحابة وبالتالي أدت إلى تحسين موازنة الحمل في السحابة.

الكلمات المفتاحية: الحوسبة السحابية (Cloud Computing)، المنصة (Eucalyptus)، المصرف التجاري السوري (Commercial Bank of Syria)، متحكم السحابة (Cloud Controller)، متحكم العنقود (Cluster Controller)، متحكم العقدة (Node Controller)، موازنة الحمل (Load Balance).

كلمة شكر

في نهاية هذا العمل لا يسعني إلا أن أتوجه بالشكر الجزيل مع فائق الاحترام والتقدير إلى جامعة دمشق لما تقدمه من إمكانيات وأساليب تساعد على المضي قدماً نحو الأمام في مجال البحث العلمي ومسيرة التطوير والتحديث.

كما أتوجه بالشكر إلى جميع المدرسين في كلية الهندسة الميكانيكية والكهربائية وأخص بالشكر السيد مشرف هذه الرسالة الدكتور أغيد القطعان، لما أبداه من اهتمام كبير وتوجيه ومساعدة ساهمت في إنجاز وإتمام هذا العمل.

كما أتوجه بالشكر لجميع العاملين في المصرف التجاري السوري وأخص بالشكر الأستاذ فراس سليمان المدير العام للمصرف التجاري السوري والسيد المهندس إياد سليمان مدير مديرية التقنية في المصرف التجاري السوري لما قدماه من مساعدة حقيقية لإنجاز هذا العمل.

الإهـــداء

***إلى الذين علماني الصبر والمثابرة وكانا شعاع المحبة والأمل... أبي وأمي الغاليين**

*** إلى رفيقة عمري ومرآة حياتي..... رزان الحبيبة**

*** إلى من تحمل غيابي لإنجاز هذا العمل..... أحمد وآدم**

***إلى من أفرح عندما ألقاهم وأحزن عندما أفارقهم ..... ذوالفقار وسارة**

*** إلى أصدقائي المخلصين ورفاقي الأحباء..... د. حافظ، علاء، سوار، حيدر**

*** إلى أقاربي وأهلي.....**

*** إلى كل من ساهم في إنجاز هذا العمل.....**

الفهرس

الملخص ..... - 2 -

كلمة شكر ..... - 3 -

الإهـداء ..... - 4 -

قائمة الأشكال ..... - 9 -

قائمة الجداول ..... - 11 -

قائمة المصطلحات ..... - 12 -

الفصل الأول ..... - 14 -

مقدمة ..... - 14 -

1-1-مقدمة ..... - 14 -

1-2- أهمية البحث ..... - 14 -

1-4- خطوات البحث ..... - 15 -

1-5- الدراسات المرجعية ..... - 15 -

1-6-تنظيم الأطروحة ..... - 25 -

1-7-الخاتمة ..... - 26 -

الفصل الثاني ..... - 28 -

أساسيات الحوسبة السحابية ..... - 28 -

2-1-مقدمة ..... - 28 -

2-2-مفهوم الحوسبة السحابية ..... - 28 -

2-3-تعريف الحوسبة السحابية ..... - 29 -

2-3-1-الخصائص الجوهرية للحوسبة السحابية ..... - 31 -

2-3-2-نماذج الخدمة ..... - 31 -

2-3-3- نماذج نشر السحابة ..... - 33 -

2-4-الحوسبة السحابية مقابل الخدمات السحابية ..... - 34 -

2-5-مميزات الحوسبة السحابية ..... - 35 -
2-6-معايير الحوسبة السحابية ..... - 35 -
2-7-منصات الحوسبة السحابية..... - 36 -
2-8-التسعير ..... - 37 -
2-9-المكونات الرئيسية للسحابة..... - 38 -
2-10-مثال على نشر تطبيق ويب ..... - 38 -
2-11-تحديات الحوسبة السحابية ..... - 40 -
2-11-1-الأداء..... - 40 -
2-11-2-الأمن والخصوصية..... - 40 -
2-11-3-التحكم ..... - 40 -
2-11-4-تكاليف معدل نقل البيانات ..... - 40 -
2-11-5-الدقة والموثوقية..... - 40 -
2-12-الحوسبة السحابية في المستقبل ..... - 41 -
2-13-الخاتمة ..... - 41 -
الفصل الثالث ..... - 43 -
موازنة الحمل في الحوسبة السحابية ..... - 43 -
3-1-مقدمة ..... - 43 -
3-2-موازنة الحمل ..... - 43 -
3-3-مقاييس تقنيات موازنة الحمل..... - 44 -
3-4-سياسة خوارزميات موازنة الحمل ..... - 44 -
3-5- الأهداف الرئيسية لخوارزمية موازنة الحمل ..... - 45 -
3-6-خوارزميات موازنة الحمل الستاتيكية المستخدمة حالياً ..... - 45 -
3-6-1-موازنة الحمل الدّوارة (Round-Robin Load Balancer) ..... - 45 -
3-6-2-MIN-MIN ..... - 46 -

3-6-3 MAX-MIN ..... - 46 -
3-6-4-موازن الحمل (Min-Min) ..... - 47 -
3-6-5-خوارزمية موازنة الحمل Max-Min-Max ..... - 47 -
3-7- خوارزميات موازنة الحمل الديناميكية المستخدمة حالياً ..... - 48 -
3-7-1-النشر المتساوي للتنفيذ الحالي (Equally Spread Current Execution)..... - 48 -
3-7-2-خوارزمية موازن الحمل المخنوق (Throttled Load Balancer)..... - 48 -
3-7-3-الحشد الفعال (Active Clustering) ..... - 48 -
3-7-4-الانضمام إلى الصف الخامل (Join-Idle Queue) ..... - 48 -
3-8- الخاتمة ..... - 49 -
الفصل الرابع ..... - 51 -
الحوسبة السحابية والآلات الافتراضية ..... - 51 -
4-1- تقنية الافتراضية (Virtualization)..... - 51 -
4-2- تطبيق تقنية الافتراضية في العتاد الصلب ..... - 51 -
4-2-1-أسباب تطبيق تقنية الافتراضية في العتاد الصلب ..... - 52 -
4-2-2-أنماط تقنية الافتراضية في العتاد الصلب ..... - 53 -
4-3- اللقطات (snapshots)..... - 54 -
4-4-التهجير (Migration) ..... - 55 -
4-5- تجاوز الفشل (Failover)..... - 55 -
4-6-الافتراضية (virtualization) والحوسبة السحابية (Cloud Computing)..... - 55 -
4-7-البنية الأساسية لتقنية الافتراضية في الحوسبة السحابية..... - 56 -
الفصل الخامس ..... - 58 -
الآلية المقترحة لتحسين موازنة الحمل ..... - 58 -
5-1-مقدمة ..... - 58 -
5-2-المنصة Eucalyptus ..... - 58 -

5-3- خوارزميات جدولة نشر الآلات الافتراضية في سحابة (Eucalyptus) ..... - 60 -
5-3-1-الخوارزمية الشرهة (Greedy Algorithm)..... - 60 -
5-3-2-خوارزمية البيان الدوار (Round Robin Algorithm) ..... - 61 -
5-4-آلية موازنة حمل نشر الآلات الافتراضية المقترحة ..... - 61 -
5-5-تصميم النظام وتحليل الأداء ..... - 66 -
5-5-1-تطبيق النظام ..... - 66 -
5-5-2-محاكاة النظام..... - 67 -
5-5-3-تحليل الأداء ..... - 69 -
5-5-4-النتائج التحليلية هي كالتالي ..... - 69 -
5-6-الخاتمة ..... - 73 -
الفصل السادس ..... - 75 -
النتائج والمقترحات والعمل في المستقبل..... - 75 -
6-1-النتائج ..... - 75 -
6-2-آفاق العمل المستقبلي ..... - 76 -
الملحق -A- ..... - 77 -
الملحق -B- ..... - 99 -
المراجع ..... - 100 -
المقالات المنشورة..... - 100 -

قائمة الأشكال

الشكل (1-1) بنية النظام المقترح ..... - 16 -
الشكل (1-2) المخطط المنطقي للخوارزمية الدّوارة الموزونة ..... - 17 -
الشكل (1-3) بنية السحابة المقترحة ..... - 20 -
الشكل (1-4) آلية عمل موازن الحمل ..... - 21 -
الشكل (1-5) الجدولة المعتمدة على الأفضلية ..... - 21 -
الشكل (1-6) بنية نظام EUQOS ..... - 22 -
الشكل (1-7) وظائف وحدة موازن الحمل ..... - 23 -
الشكل (1-8) إجرائية تشغيل موازن الحمل ..... - 24 -
الشكل (1-9) سجل خدمة المعالجة ..... - 25 -
الشكل (2-1) ست نماذج حوسبة ..... - 29 -
الشكل (2-2) نموذج (NIST) للحوسبة السحابية ..... - 30 -
الشكل (2-3) نماذج الخدمة التي يوفرها مزود السحابة ..... - 32 -
الشكل (2-4) مثال على نشر تطبيق ..... - 38 -
الشكل (2-5) بنية سحابة هجينة موزعة ..... - 41 -
الشكل (4-1) البنية الأساسية لتقنية الافتراضية في الحوسبة السحابية ..... - 56 -
الشكل (5-1) بنية سحابة EUCALYPTUS ..... - 60 -
الشكل (5-2) البنية التنظيمية لسحابة EUCALYPTUS ..... - 62 -
الشكل (5-3) المخطط التدفقي لاختيار NC مناسبة لتشغيل الآلة الافتراضية عليها ..... - 64 -
الشكل (5-4) توضيح مبسط لإجرائية تشغيل الالة الافتراضية ..... - 65 -
الشكل (5-7) عمليات متحكم مراقبة حالة الشبكة المضاف ..... - 68 -
الشكل (5-8) توضع الآلات الافتراضية مع آليات الجدولة المختلفة للحالة الدراسية الأولى ..... - 70 -
الشكل (5-9) مخطط وقت اكتمال تشغيل كل VM في الحالة الدراسية الأولى ..... - 70 -
الشكل (5-11) مخطط وقت اكتمال تشغيل كل VM في الحالة الدراسية الثانية ..... - 72 -
الشكل (A-1) الإقلاع من قرص UBUNTU ..... - 78 -
الشكل (A-2) اختيار الشبكة ETH0 ..... - 78 -
الشكل (A-3) اختيار اسم متحكم السحابة ..... - 79 -
الشكل (A-4) الاستمرار إلى الشاشة التالية ..... - 79 -
الشكل(A-5) اختيار متحكم السحابة ..... - 80 -

الشكل (A-6) استخدم واجهة ETH0 للاتصال بالشبكة العامة..... - 81 -
الشكل (A-7) خيارات تقسيم القرص الصلب ..... - 81 -
الشكل (A-8) اختيار القرص الصلب..... - 82 -
الشكل (A-9) رسالة تتطلب الموافقة على حذف كافة المعلومات من القرص الصلب ..... - 82 -
الشكل (A-10) الموافقة على التقسيم الافتراضي للقرص الصلب ..... - 83 -
الشكل (A-11) استخدام كامل مساحة القرص الصلب ..... - 83 -
الشكل (A-12) حفظ التغيرات الجديدة على القرص الصلب ..... - 84 -
الشكل (A-13) تهيئة القرص الصلب وتنصيب نظام التشغيل ..... - 84 -
الشكل (A-14) إدخال اسم المستخدم CLADMIN وكلمة المرور CLOUD9 ..... - 85 -
الشكل (A-15) عدم القبول بالتحديثات..... - 85 -
الشكل (A-16) كتابة اسم العنقود ..... - 86 -
الشكل (A-17) إدخال مجموعة عناوين IPS..... - 86 -
الشكل (A-18) تنصيب GRUB ..... - 87 -
الشكل(A-19) الانتهاء من تنصيب متحكم السحابة..... - 87 -
الشكل(A-20) تنصيب نظام تشغيل UBUNTU ..... - 88 -
الشكل(A-21) إدخال اسم العنقود..... - 88 -
الشكل(A-22) اختيار متحكم العنقود..... - 89 -
الشكل(A-23) اختيار متحكم عنقود آخر موجود في الشبكة..... - 89 -
الشكل(A-24) واجهة الدخول إلى السحابة..... - 91 -
الشكل(A-25) واجهة تغيير كلمة المرور ..... - 91 -
الشكل(A-26) تحميل الاعتمادية..... - 92 -
الشكل(A-27) صفحة IMAGE ..... - 93 -
الشكل(A-28) تنصيب الـ IMAGE..... - 94 -
الشكل(A-29) صفحة الـ IMAGE..... - 94 -
الشكل (A-30) استعراض العنوان للحالة..... - 97 -
الشكل (B-1) واجهة برنامج مراقب حالة الشبكة..... - 99 -

قائمة الجداول

الجدول	الصفحة
الجدول–1-1– زمن الاستجابة الكلي لخوارزمية (RR)	17
الجدول –1-2– زمن المعالجة لمركز بيانات السحابة من اجل خوارزمية (RR)	17
الجدول –1-3– زمن الاستجابة الكلي لخوارزمية (WRR)	17
الجدول –1-4– زمن المعالجة لمركز بيانات السحابة من اجل خوارزمية (WRR)	18
الجدول–1-5–زمن التنفيذ المتوسط	24
الجدول –2-1–السمات الأساسية للحوسبة السحابية	33
الجدول –2-2–السمات الأساسية للخدمات السحابية	33
الجدول –2-3–معايير الحوسبة السحابية	35
الجدول –2-4–مزودي منصات الحوسبة السحابية	36
الجدول –2-5– مقارنة بين أسعار منصات الحوسبة السحابية الرئيسية	36
الجدول –2-6– الشركات الرائدة في مجال مكونات الحوسبة السحابية	37
الجدول –2-7– أمثلة الحوسبة السحابية في الشركات الكبيرة	38

قائمة المصطلحات

Availability	المتاحية
Cloud Computing	الحوسبة السحابية
Cloud Storage	التخزين السحابي
Confidentiality	السريّة
Encryption	التشفير
Fairness / Waiting Time	زمن الانتظار
Fault tolerance	عدم الفشل
File Sharing	مشاركة الملفات
Infrastructure as a Service (IaaS)	البنية التحتية كخدمة
Integrity	الأمانة
Load balancing	موازنة الحمل
Network Attached Storage	خازن متصل بالشبكة
Node Controller (NC)	متحكم العقدة
Performance	الأداء
Platform as a Service (PaaS)	المنصة كخدمة
Resource utilization	استخدام الموارد
Response time	زمن الاستجابة
Scalability	قابلية التوسع
Software as a Service (SaaS)	البرمجيات كخدمة
Synchronization	المزامنة
Throughput	سعة المعالجة
Two Factor Authentication	ميزة التحقق بخطوتين
Virtual Machines [VMs]	الآلات الافتراضية

الفصل الأول

مقدمة

الفصل الأول
مقدمة

1-1-مقدمة

السحابة هو تعبير كان يُستخدم في البداية للإشارة إلى الإنترنت وذلك في مخططات الشبكات، حيث عُرف على أنه رسم أولي لسحابة يتم استخدامها لتمثيل نقل البيانات من مراكز البيانات إلى موقعها النهائي في الجانب الآخر من السحابة، وقد بدأت الفكرة الكامنة وراء الحوسبة السحابية من خلال طرح فكرة البرامج كخدمات عندما عبر (جون مكارثي)[1] الأستاذ بجامعة ستاندفورد عن الفكرة بقوله " قد تنظم الحوسبة لكي تصبح خدمة عامة في يوم من الأيام " حيث رأى أنه من الممكن أن تؤدي تكنولوجيا مشاركة الوقت (Time Sharing) إلى مستقبل تُباع فيه الطاقة الحاسوبية، وأصبحت هذه الفكرة مصطلحاً شائعاً في الدوائر التكنولوجية والمؤسسات في وقتنا الحالي.

المتاحية والموثوقية التي توفرها السحابة أمر هام بالنسبة لكافة المستخدمين. وفي الآونة الأخيرة تحول سلوك المستخدم إلى استخدام أنواع مختلفة من الأجهزة منها أجهزة الهواتف الذكية والأجهزة المحمولة والأجهزة اللوحية، وبما أن الوصول إلى السحابة سيتم من أماكن مختلفة ومن قبل أنواع مختلفة من الأجهزة والتجهيزات سيؤدي ذلك إلى وجود اختلاف في حالة كل اتصال بالشبكة مع السحابة وستكون حالة هذا الاتصال أحياناً غير متوقع.

1-2- أهمية البحث

تقدم السحابة الخدمات للمستخدمين عن طريق نشر آلات افتراضية (Virtual Machines [VMs]) ضمن بنيتها التحتية، وتعد طريقة نشر الآلات الافتراضية (VMs) ضمن بنيان الحوسبة السحابية (Cloud Computing Architecture) أمر حاسم للغاية للحفاظ على الجودة المتوقعة من خدمات السحابة. عملياً كل مستخدم مرتبط بآلة افتراضية واحدة أو أكثر موزعة ضمن مركز بيانات السحابة، وبالتأكيد تختلف طوبولوجية الشبكة للحوسبة السحابية باختلاف البنية التحتية للسحابة.

إنّ موازنة الحمل في الأنظمة السحابية تعد قضية حاسمة لتحقيق الاستخدام الأمثل للموارد وتحسين زمن استجابة النظام والحفاظ على خدمة متاحة بشكل دائم وبالتالي تحسين أداء النظام، ينظر هذ البحث في بعض آليات نشر الآلات الافتراضية والتي تلعب دوراً هاماً في موازنة حمل

الأنظمة السحابية. هدفنا هو تحسين جودة وكفاءة أداء السحابة الخاصة من خلال تحسين طوبولوجيا نشر الآلات الافتراضية.

1-3-الهدف من البحث

يهدف هذا البحث إلى تحسين جودة وكفاءة أداء السحابة الخاصة من خلال تحسين طوبولوجيا نشر الآلات الافتراضية، والتي بدورها تلعب دوراً مهماً في تحقيق حمل متوازن ومستقر في السحابة لذلك تم دراسة وتحليل تقنيات وآليات نشر الآلات الافتراضية في الحوسبة السحابية وتم اقتراح وتطبيق آلية جديدة لنشر الآلات الافتراضية حيث قمنا بإضافة متحكم جديد (تطبيق برمجي) إلى بنية السحابة مهمته مراقبة حالة الشبكة (بين متحكمات السحابة والعقد من جهة وبين متحكمات السحابة والمستخدمين من جهة أخرى) بشكل مستمر ليقوم بتحديد الطوبولوجيا الأفضل لنشر الآلات الافتراضية.

1-4- خطوات البحث

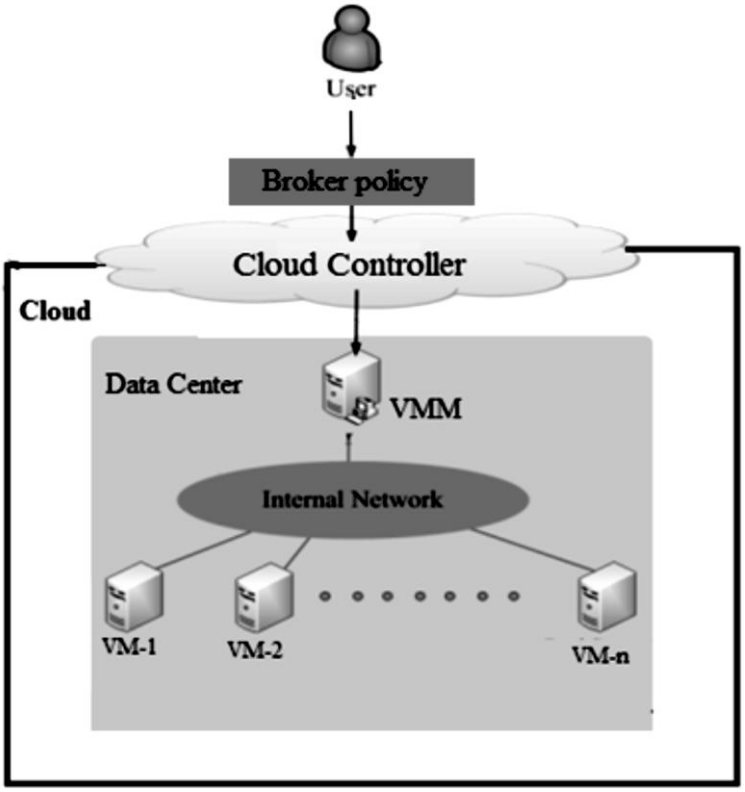
- دراسة البنية التحتية للحوسبة السحابية ومكوناتها.
- دراسة بعض الخوارزميات السابقة لموازنة الحمل.
- دراسة مفهوم الافتراضية.
- دراسة بعض منصات الحوسبة السحابية ومنها سحابة Eucalyptus مفتوحة المصدر.
- اقتراح آلية جديدة لنشر الآلات الافتراضية لتحسين موازنة الحمل في سحابة Eucalyptus ومقارنة نتائجها مع بعض الخوارزميات السابقة.

1-5- الدراسات المرجعية

ظهر دور الآلات الافتراضية (VMs) كقضية مهمة في موازنة الحمل، والتي تجعل البنى التحتية للحوسبة السحابية قابلة للتوسع. ولتحسين الأداء العامّ لاستخدام الحوسبة السحابية، يجب أن تدار الآلات الافتراضية (VMs) بعناية لكي تعمل بشكل جيد وتكون موزّعة بشكل صحيح بين الموارد الفيزيائية. نستعرض بعض الدراسات المرجعية السابقة التي عالجت مشكلة آلية نشر الآلات الافتراضية (VMs) في سحابة (Eucalyptus).

1- قام (Supreeth S, Shobha Biradar, 2013)[2] بنشر ورقة بحث بعنوان جدولة الآلات الافتراضية لموازنة الحمل في منصات الحوسبة السحابية (Scheduling Virtual Machines for

Load balancing in Cloud Computing Platform) تقترح هذه النظرية نظاماً لتحسين أداء الآلات الافتراضية في بناء السحابة وذلك لتحسين قدرة الآلات الافتراضية (VMs) على تقديم الخدمات الموثوقة في منصات الحوسبة السحابية. يُظهر الشكل (1-1) بنية النظام والذي يشمل متحكم السحابة (CLC) إن متحكم السحابة هو نقطة الدخول إلى النظام وهو المسؤول عن إدارة الموارد الافتراضية التحتية من مخدمات ومخازن وشبكات. إنّ متحكم السحابة هو مجموعة الخدمات المجمّعة بالأصناف الثلاثة: خدمات المصدر وخدمات البيانات وخدمات الواجهة، ويعالج أيضاً بروتوكولات النقل.

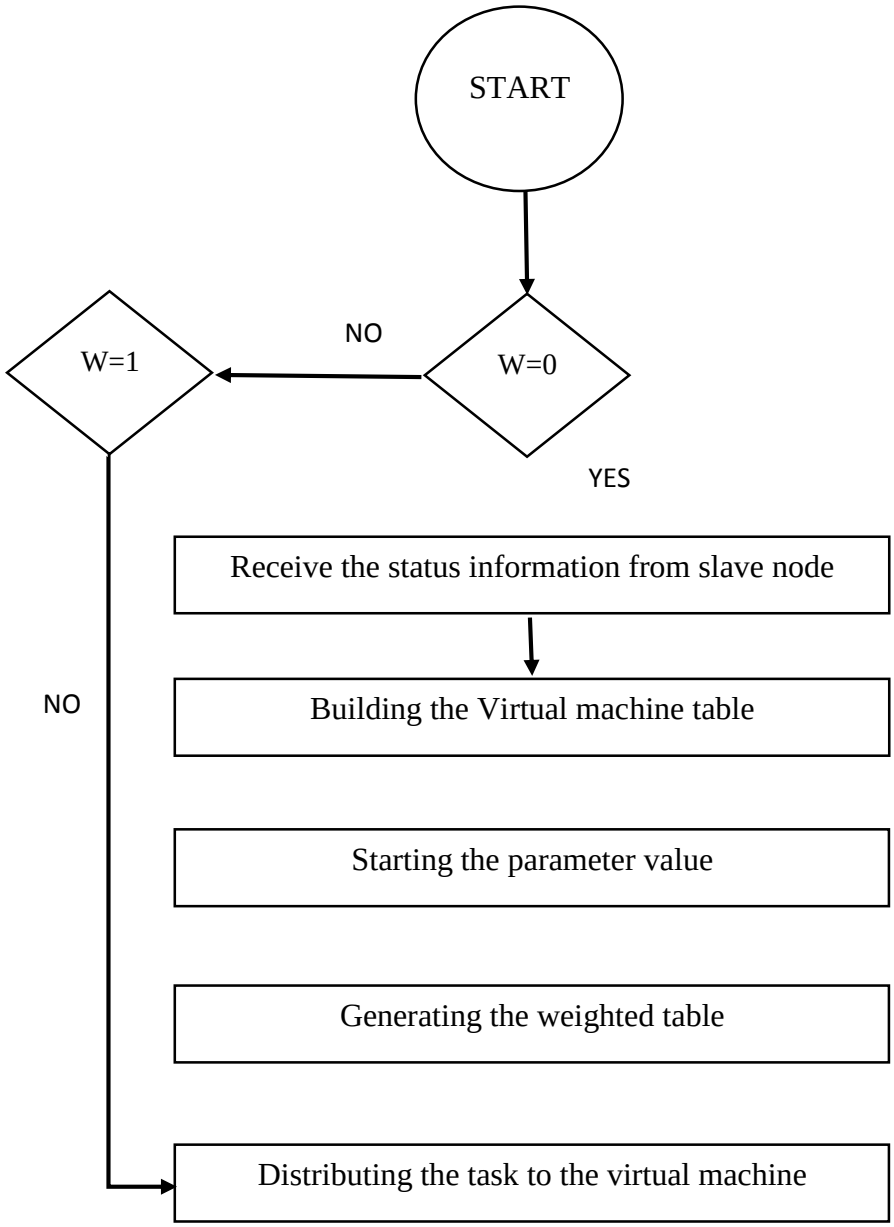


الشكل (1-1) بنية النظام المقترح

ويتم في هذه الخوارزمية تخصيص كلّ الطلبات القادمة إلى الآلات الافتراضية المتوفرة في الخوارزمية الدّوارة المستندة على الوزن بدون الأخذ بعين الاعتبار الحمل الحالي على كلّ آلة افتراضية، ويبين الشكل (2-1) المخطط المنطقي للخوارزمية الدّوارة الموزونة.

وتستخدم الخوارزمية المقترحة محاكاة تخطيطية وتم استخدام لغة جافا لتطبيق خوارزمية موازنة الحمل للآلات الافتراضية. مع الفرض إنّ التطبيق ينشر في مركز بيانات واحد ويملك الآلات الافتراضية

ذات الخصائص التالية (2048 ميغابايت حجم الذاكرة لكلّ آلة افتراضية والتي تعمل على معالج فيزيائي تصل سرعته إلى 100 MIPS). وتُظهر النتائج عند تطبيق الخوارزمية الدّوارة (Round Robin (RR)) زمن الاستجابة الكلي للسحابة، عند الزمن الأصغري (*Min (ms)*) والزمن الأعظمي (*Max (ms)*) بالنسبة لعدة آلات افتراضية تم تحليلها والمبينة في الجدول–1-1–.



الشكل (2-1) المخطط المنطقي للخوارزمية الدّوارة الموزونة

الجدول–1-1– زمن الاستجابة الكلي لخوارزمية (RR)

# of VM's	Avg (ms)	Min (ms)	Max (ms)
10	307.94	271.64	346.648
20	304.686	264.265	369.265
30	303.48	240.389	369.39
40	302.304	240.514	369.515
50	301.286	240.639	369.64

وتُظهر النتائج أيضاً زمن المعالجة لمركز بيانات السحابة عند تطبيق الخوارزمية الدّوارة (Round Robin (RR))، عند الزمن الأصغري (*Min (ms)*) والزمن الأعظمي (*Max (ms)*) بالنسبة لعدة آلات افتراضية تم تحليلها والمبينة في الجدول –2-3–.

الجدول –2-1– زمن المعالجة لمركز بيانات السحابة من اجل خوارزمية (RR)

# of VM's	Avg (ms)	Min (ms)	Max (ms)
10	.43	.127	.635
20	.526	.209	.76
30	.604	.309	.885
40	.733	.409	1.01
50	.846	.509	1.135

كما تظهر النتائج المستندة على الخوارزمية الدّوارة الموزونة (Weighted Round Robin (WRR)) زمن الاستجابة الكلي في السحابة، عند الزمن الأصغري (*Min (ms)*) والزمن الأعظمي (*Max (ms)*) بالنسبة لعدة آلات افتراضية تم تحليلها والمبينة في الجدول –1-3–.

الجدول –3-1– زمن الاستجابة الكلي لخوارزمية (WRR)

# of VM's	Avg (ms)	Min (ms)	Max (ms)
10	300.08	237.069	369.14
20	300.2	237.119	369.265
30	300.32	237.169	369.39
40	300.44	237.219	369.515
50	300.56	237.269	369.64

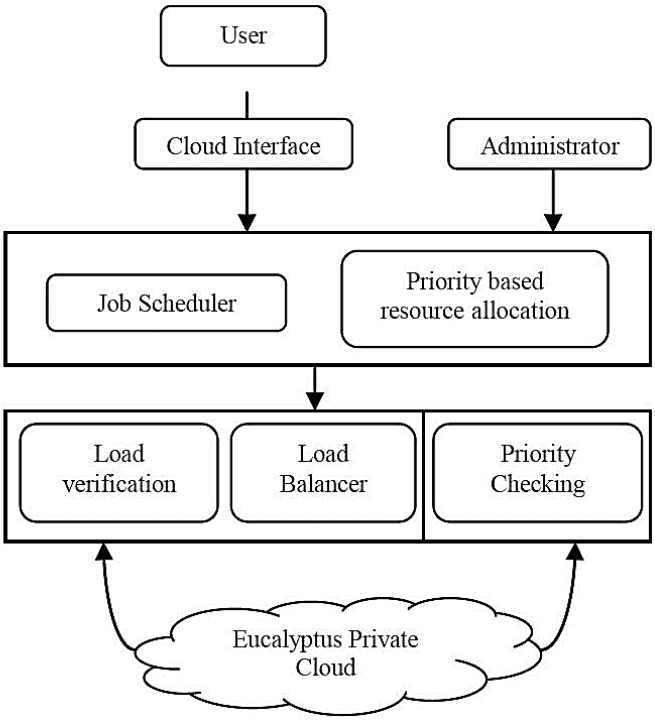
وتُظهر النتائج أيضاً زمن المعالجة لمركز بيانات السحابة عند تطبيق الخوارزمية الدّوارة الموزونة (Weighted Round Robin (RR))، عند الزمن الأصغري (*Min (ms)*) والزمن الأعظمي (*Max (ms)*) بالنسبة لعدة آلات افتراضية تم تحليلها والمبينة في الجدول –4-1–.

الجدول –4-1– زمن المعالجة لمركز بيانات السحابة من اجل خوارزمية (WRR)

# of VM's	*Avg (ms)*	*Min (ms)*	*Max (ms)*
10	.366	.022	.637
20	.486	.034	.762
30	.606	.047	.887
40	.726	.059	1.012
50	.836	.072	1.137

يظهر من مقارنة نتائج التجربة في الأعلى أن الخوارزمية الدّوارة الموزونة تستهلك زمن استجابة اقل من الخوارزمية الدّوارة.

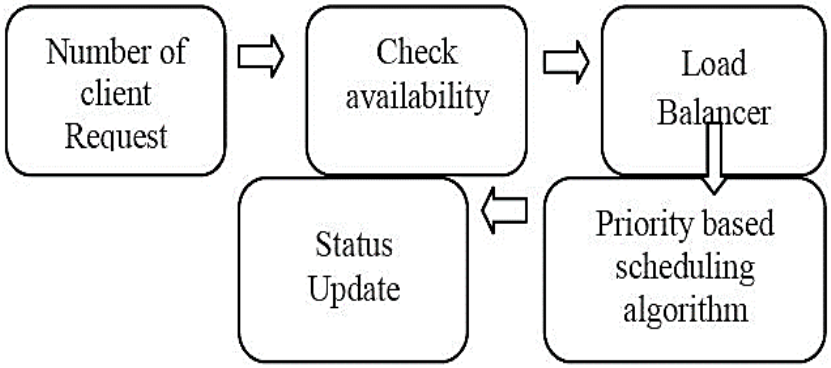
2– وكما قام (G.Suryadevi, D.Vijayakumar, R.SabariMuthuKumar, Dr. K .G. Srinivasagan, 2014) [3] بنشر دراسة خوارزمية موازنة الحمل الديناميكية للتوزيع الكفوء في بيئة السحابة الخاصة تحت عنوان (An Efficient Distributed Dynamic Load Balancing Algorithm for Private Cloud Environment) وتقترح هذه الخوارزمية بنية نظام بحيث يمكن تخصيص الموارد لموازنة الحمل في سحابة (Eucalyptus) الخاصة حيث تتألف سحابة (Eucalyptus) من خمسة مكونات: متحكم السحابة (Cloud Controller (CLC))، متحكم (Walrus)، متحكم العنقود (Cluster Controller (CC))، متحكم التخزين (Storage Controller (SC)) ومتحكم العقدة (Node Controller(NC)) بالإضافة إلى الخيار الإضافي (VMware Broker)، كل مكون من هذه المكونات مستقل كخدمة ويب ما عدا (VMware Broker). بينما تتألف البنية المقترحة في هذا البحث من ثلاثة وحدات: الوحدة الأولى هي مجدول العمل (Job Scheduler) حيث يرسل الزبون طلباته إلى المخدم والتي يتم تقديمها في طبقة البنية التحتية كخدمة (Infrastructure as a Service (IaaS)) في بيئة السحابة وبالتالي يتم أولاً اختبار الموارد المتوفرة في السحابة من قبل مجدول العمل (Job Scheduler) ويقوم بدوره بتسجيل جدولة التنفيذ لكل الموارد باستخدام (slot) ويختبر توافر الموارد في السحابة، ويبين الشكل (3-1) بنية السحابة المقترحة.



الشكل (3-1) بنية السحابة المقترحة

الوحدة الثانية هي مخصص الموارد (Resource Allocation) يتم في هذه الوحدة مباشرةً توجيه الطلبات الواردة إلى الآلة الافتراضية. والوحدة الثالثة هي موازن الحمل والذي يسمح بتوجيه الطلبات الواردة إلى الخدمات التي تستضيف التطبيق.

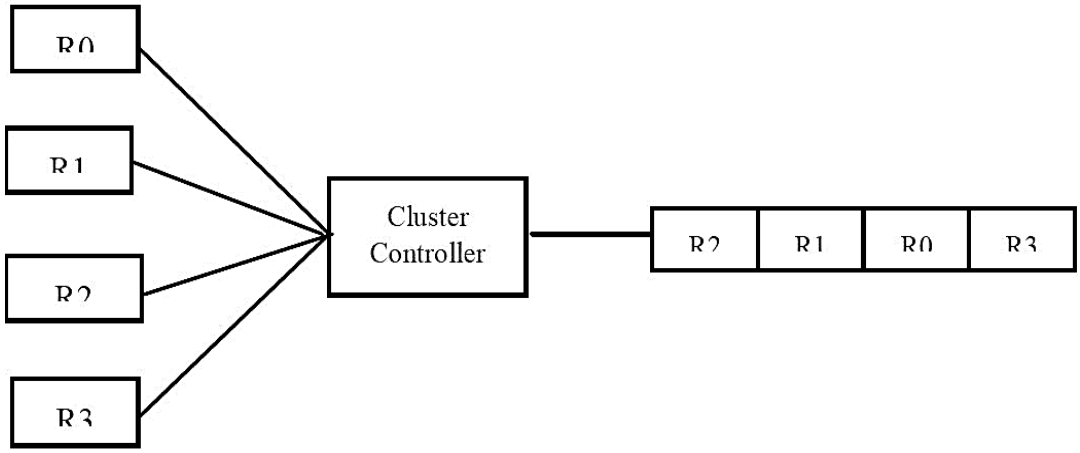
وتعتمد آلية عمل موازن الحمل على إنشاء تطبيق ويب في موازن الحمل بحيث يسمح للعديد من الزبائن بالدخول إلى الآلة الافتراضية بفعالية من جهة الزبون. أولياً يرسل العديد من الزبائن المهمة إلى المخدم. ويتم وضع طلبات الزبائن في الصف وترسل إلى موازن الحمل. ويختبر موازن الحمل الموارد الافتراضية المتوفرة المتصلة به، ولديه معلومات عن حالة كل الآلات الافتراضية المتصلة معه ما إذا كانت هذه الآلة مشغولة أو في حالة متوفرة. واستناداً على حالة الآلة الافتراضية يمكن تخصيص الموارد الافتراضية للمهمة. ومن أجل اختبار توافر الآلة الافتراضية يمكن تحديث حالة الآلة الافتراضية كما هو مبين في الشكل (4-1).



الشكل (4-1) آلية عمل موازن الحمل

ويتم تمثيل تحديث الحالة على شكل جدول لتمثيل حالة سرعة المعالج (CPU) والذاكرة والقرص الصلب لمختلف الآلات الافتراضية. ويُستخدم تحديث الحالة لتمثيل توافر الآلات الافتراضية فيما إذا كانت مشغولة أو في حالة خاملة، ويتم تخصيصها لمهمة إذا كانت حرة، وبالتالي يكون ناتج آلية عمل موازن الحمل استخدام الموارد المتوفرة بشكل كفوء، وعندما يكون الطلب القادم ذو حمل إضافي عندها يقوم المدير (Admin) بتحديد الأفضلية لكل طلب. ترسل العديد من الطلبات إلى المخدم وتوضع الأفضلية لكل طلب وتستند الأفضلية على عامل السعة والحمل وباستخدام الأفضلية يتم توجيه الطلبات إلى العقد.

وتقوم خوارزمية الأفضلية الديناميكية (Dynamic Priority Based Algorithm) بموازنة الحمل الديناميكية للعقد عند توزيع الحمل، وتستخدم خوارزمية الجدولة المعتمدة على الأفضلية من قبل المنظمات التي تحتاج إلى تطبيق حوسبة سحابية صغيرة إلى متوسطة الحجم وتعمل هذه الخوارزمية على مبدأ (FIFO) كما هو مبين في الشكل (5-1).



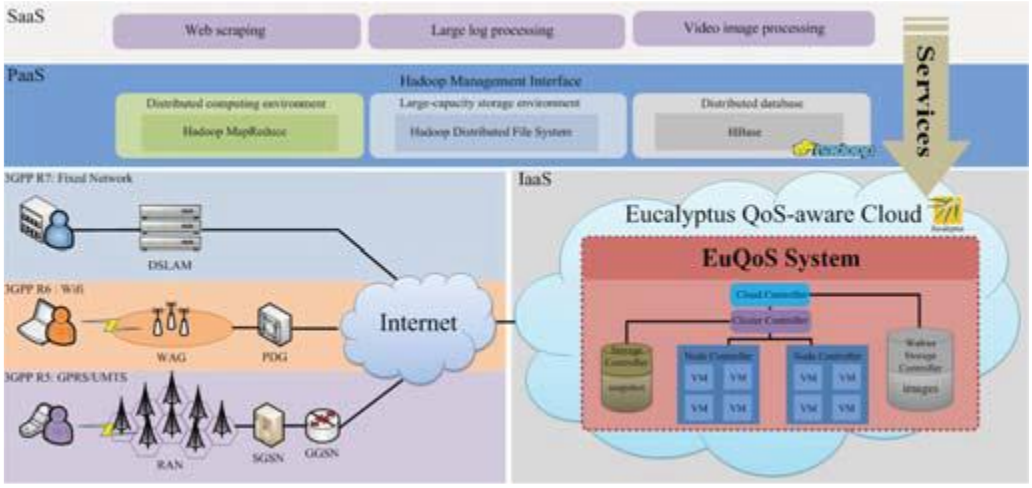
الشكل (5-1) الجدولة المعتمدة على الأفضلية

والفائدة من استخدام موازنة الحمل الديناميكية هو في حال فشل أية عقدة عندها لا يتوقف تشغيل النظام ولكن يتأثر أداء النظام فقط. وعند المقارنة مع البيئة المركزية في نظام موازنة الحمل الديناميكية الموزعة، يمكن أن تعمل العقد مع بعضها البعض، علاوة على ذلك إن اختيار المخدم المناسب يحتاج إلى اتصال في الزمن الحقيقي مع العقد الأخرى في الشبكة. ولذلك يتم توليد العديد من الرسائل في الشبكة، ويستخدم موازن الحمل الديناميكي سياسات موازنة الحمل لمراقبة تحديث المعلومات.

ونلاحظ من دراسة نتائج هذا البحث أنه تم تقديم خوارزمية جديدة في بيئة (Eucalyptus) لجدولة الآلات الافتراضية (خوارزمية موازنة الحمل الديناميكية للتوزيع الكفوء في بيئة السحابة الخاصة) ويمكن تطوير هذه الخوارزمية حتى تكون مناسبة لبيئات أخرى، هنا كل الطلبات الواردة من الزبائن يتم توجيهها أوتوماتيكياً إلى الآلات الافتراضية باعتماد قيمة الأفضلية، وبالتالي يتم توليد آلات افتراضية جديدة وإعادة توجيه الطلبات عندما يصبح هناك حملاً زائداً. وتظهر النتائج التجريبية الاستخدام الأفضل للآلات الافتراضية وتخفيض زمن التنفيذ.

3- وكذلك قام كل من الباحثين (Jiann-Liang Chen, Yanuarius Teofilus Larosa and Pei-Jia Yang)[4] بدراسة آلية موازنة الحمل (QoS) المثالية لجدولة الآلات الافتراضية في منصة سحابة Eucalyptus.

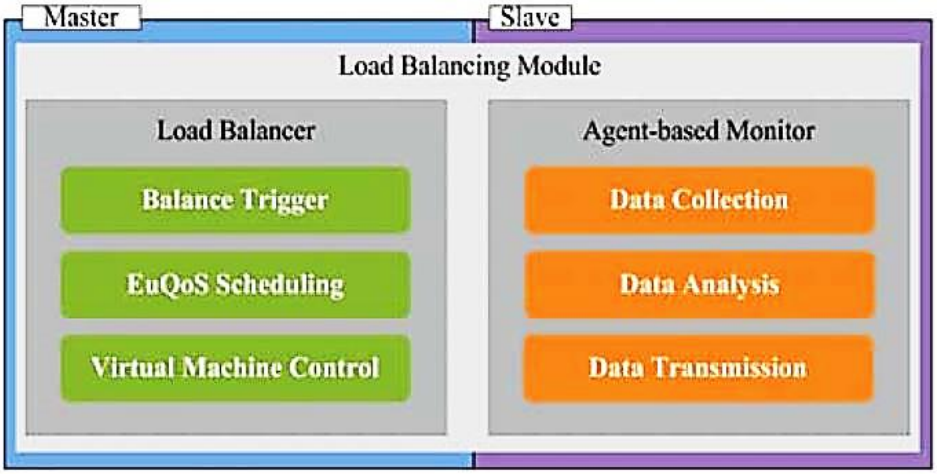
تقوم هذه الخوارزمية على تطوير آلية لموازنة الحمل بحيث نتمكن من تحسين أداء الآلات الافتراضية (VMs). تدعى نظام جودة الخدمة في Eucalyptus (EuQoS)، وذلك بتعديل منصة Eucalyptus. بحيث يتم في نظام (EuQoS) خلق وكلاء لمراقبة آلية موازنة الحمل. ويبين الشكل (6-1) بنية نظام (EuQoS).



الشكل (6-1) بنية نظام EuQoS

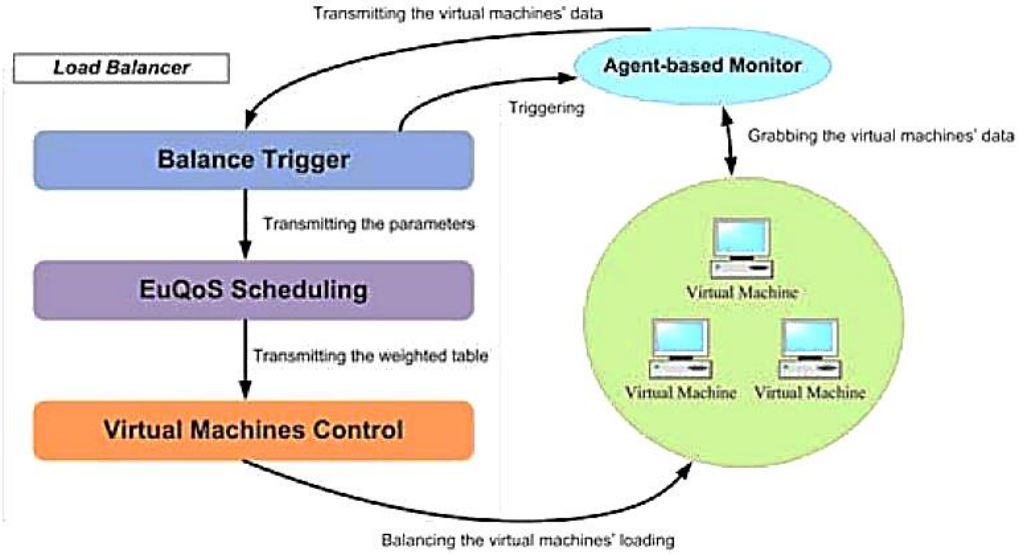
حيث يتصل المستخدمون بسحابة (Eucalyptus) ليحصلوا على الخدمات أو المعطيات. يتألف المستوى الأدنى لسحابة (Eucalyptus) من عدد من الآلات الافتراضية (VMs) التي تشمل موزع المعطيات (dispersive data). وعندما يدخل عدد كبير من المستخدمين إلى هذه الخدمات تنقص القدرة الحاسوبية لهذه الآلات الافتراضية. للتعامل مع هذه المشكلة، تدعم آلية موازنة الحمل إطار (Hadoop frameworks)، والتي تتألف من ثلاث مكونات رئيسية: نظام الملفات الموزعة لـ Hadoop (HDFS) و (MapReduce) وقاعدة بيانات Hadoop (HBase).

وتقترح الدراسة وحدة موازنة الحمل لتوزيع حمل العمل الإضافي للنظام للآلات الافتراضية في سحابة (Eucalyptus). يظهر الشكل (7-1) وظائف وحدة موازنة الحمل وتشمل مكوّنين: موازن الحمل والمراقب الوكيل. يدعم موازن الحمل ثلاث قابليات في ثلاث آليات محققة في النظام المقترح (EuQos): مشغل الموازن (balance triggering)، جدولة (EuQos) ومتحكم VM. بحيث تستقبل آلية مشغل الموازن البيانات من المراقب الوكيل. وتنفذ آلية جدولة (EuQos) طريقة الموازنة بتحليل معطيات الآلات الافتراضية، وتعطي آلية التحكم بالآلة الافتراضية متحكم الموازن حمل الآلات الافتراضية وفقاً للأعمال المأخوذة من قبل جدولة (EuQos). وينفذ المراقب الوكيل جمع المعطيات وتحليل المعطيات وقابلية نقل المعطيات. هذه القابليات مطبقة على الجدولة ومتكاملة مع المراقب الوكيل. وتنفذ وحدة جمع المعطيات مراقبة الآلات الافتراضية للحصول على بعض البارامترات المطلوبة من قبل موازن الحمل. وتقرر وحدة تحليل المعطيات فيما إذا كانت معطيات الآلات الافتراضية سوف تبقى في نفس الحالة السابقة. وتنفذ وحدة نقل المعطيات توزيع المعطيات من وحدة تحليل المعطيات إلى وحدة موازن الحمل.



الشكل (7-1) وظائف وحدة موازن الحمل

وتتم عملية موازن الحمل الشكل(8-1). عندما يبدأ مشغل موازن الحمل آليته، يستقبل بارامترات حالة الآلات الافتراضية من المراقب الوكيل ويختبر فيما إذا كانت حالة VMs لا تزال حية. إن قابلية جدولة EuQoS تمسك البارامترات وتحسبها حيث يجب أن تخزن هذه النتيجة في جدول الأوزان. ثم توزع وحدة التحكم بالآلة الافتراضية حمل الآلات الافتراضية حسب القيمة المخزنة في جدول الأوزان.



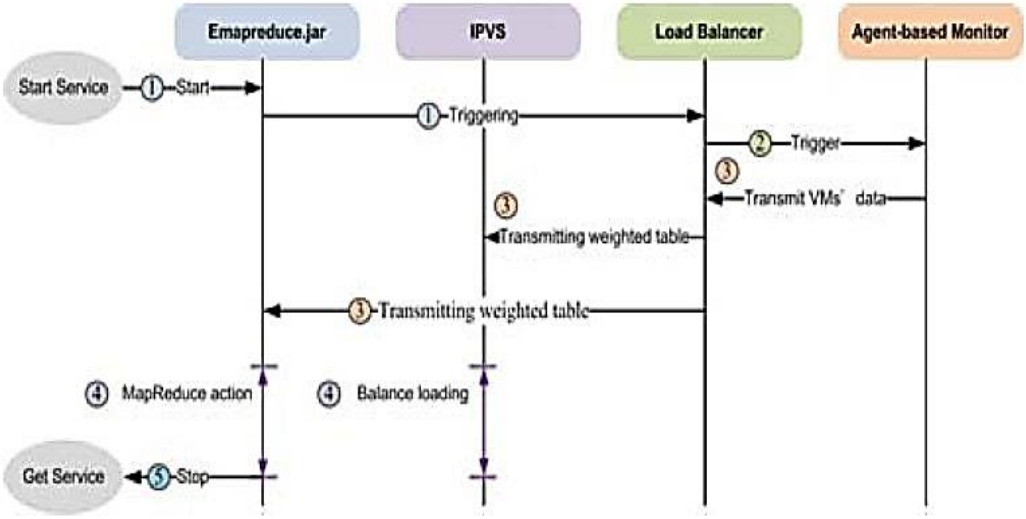
الشكل (8-1) إجرائية تشغيل موازن الحمل

-تحليل الأداء:

تم هنا مقارنة آلية EuQoS المقترحة (نظام EuQoS) بالنظام الأساسي (Eucalyptus المتضمنة مع Hadoop) والنظام الأساسي مع وحدة موازن Hadoop.

أ- بيئة المحاكاة:

في هذه الخوارزمية، يتم بناء المراقب الوكيل ضمن حالة instances لكي يمسك(يجمع)، يحلّل ويرسل الحالات إلى موازن الحمل. إنّ موازن الحمل مركّب على Instances1 (الحالة الرئيسية) يزوّد مشغل موازن الحمل، وجدولة EuQoS. وآليات التحكم بالآلات الافتراضية. في هذه الدراسة، ويقيس سجل خدمات المعالجة أداء نظام EuQoS، ويبين الشكل (9-1) أداء المعالجة من قبل سجل خدمة المعالجة.



الشكل (9-1) سجل خدمة المعالجة

ب–نتيجة ومناقشة المحاكاة في هذه الخوارزمية:

في هذه الدراسة، حجوم البيانات من 3.78 ميغابايت (قطعة 1)، 7.56 ميغابايت (قطعتان) و11.34 ميغابايت (3 قطع) تستخدم لقياس الأداء، البدء في الثانية 20. تستخدم قياس سعة المعالجة Throughput كمؤشر لمقارنة أداء Eucalyptus المتضمن Hadoop، النظام الأساسي مع Hadoop الأساسي المبني فيه، ونظام EuQoS. كلّ التجارب كرّرت 10 مرات. يبين الجدول –5-3– زمن التنفيذ المتوسط للحالات الثلاثة، يشير بأنّ زمن التنفيذ ينقص كلما زادت كمية البيانات. ربما كان سبب هذه الظاهرة تقييدات أجهزة VMs أو حجم الكتلة المحدود في Hadoop.

الجدول–5-1–زمن التنفيذ المتوسط

Cases	Average Execution Time (sec)		
	Basic System (Eucalyptus and Hadoop)	Basic System with Hadoop Balancer	EuQoS System
1 Chunk	118.23	117.56 (+0.57%)	116.32 (+1.62%)
2 Chunks	159.72	158.89 (+0.47%)	157.93 (+1%)
3 Chunks	236.56	235.78 (+0.32%)	234.67 (+0.79%)
(...): Improved ratio of execution time 1 Chunk = 3.78MB			

1–6–تنظيم الأطروحة

تم تنظيم الأطروحة بستة فصول وملحقين، فبالإضافة إلى الفصل الأول (المقدمة) يقدم الفصل الثاني فكرة عامة عن الحوسبة السحابية، مكوناتها، الوظائف والخدمات التي تقدمها السحابة. ويقدم الفصل الثالث مفهوم موازنة الحمل في النظم الموزعة، أهدافها، أنواعها والسياسات المختلفة

المستخدمة، ودرسنا بعضاً من خوارزميات موازنة الحمل السابقة التي تنطبق على السحب من مختلف الأحجام. أما في الفصل الرابع تم فيه عرض مفهوم التقنية الافتراضية، ويعرض الفصل الخامس العمل المقترح، ويقدم الفصل السادس النتائج والاستنتاجات حول الأعمال المنجزة في هذه الأطروحة والعمل المستقبلي.

1-7-الخاتمة

يعطي هذا الفصل مقدمة عن الحوسبة السحابية. وفكرة عامة عن أهداف وأهمية هذا البحث وتنظيم هذه الأطروحة.

الفصل الثاني

أساسيات الحوسبة السحابية

الفصل الثاني
أساسيات الحوسبة السحابية

2-1-مقدمة

في هذا الفصل، نقوم بتعريف مفهوم الحوسبة السحابية والخدمات السحابية، كما نُقدم طبقات وأنواع الحوسبة السحابية. وسنناقش الفرق بين الحوسبة السحابية والخدمات السحابية، أما التقنيات الجديدة التي وفرتها الحوسبة السحابية فسوف تُعرض لاحقاً. كما نناقش أيضاً مميزات الحوسبة السحابية، ومعاييرها والقضايا الأمنية المتعلقة بها. هذا بالإضافة لعرض منصات الحوسبة السحابية الرئيسية، والشركات المقدمة للخدمات السحابية. فضلاً عن مناقشة تحديات الحوسبة السحابية ومستقبلها.

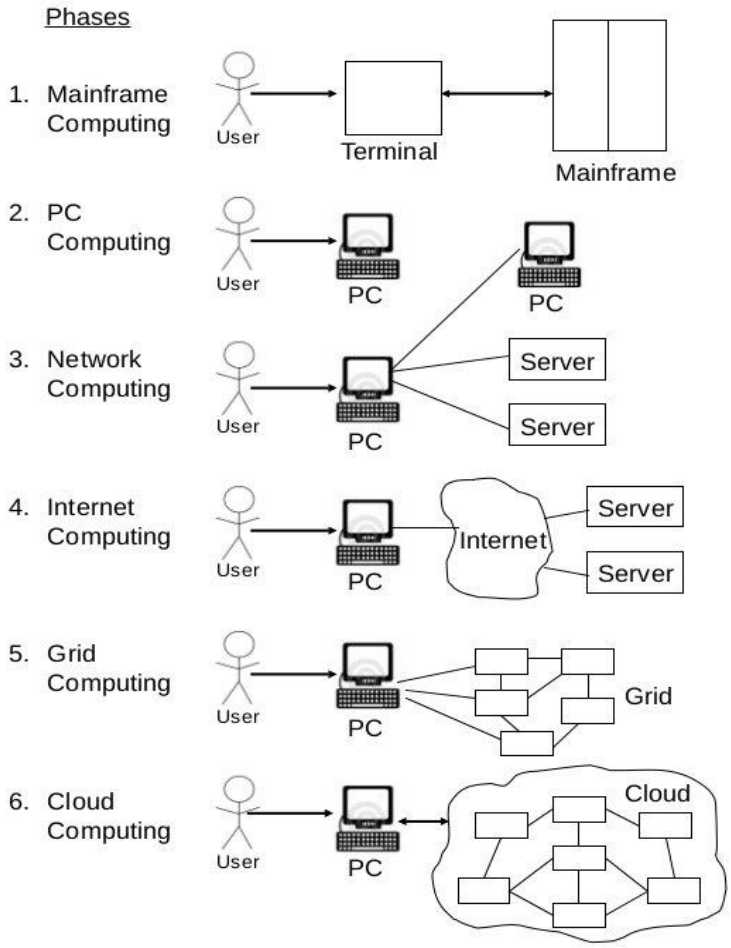
2-2-مفهوم الحوسبة السحابية

تعتبر الحوسبة السحابية نمط جديد للحوسبة يتم فيه توفير موارد قابلة للتطوير بشكل حيوي وافتراضي في كثير من الأحيان باعتبارها خدمات مقدمة عبر الإنترنت. لقد أصبحت الحوسبة السحابية اتجاهاً تقنياً هاماً، ويتوقع العديد من الخبراء أن الحوسبة السحابية ستعيد تشكيل عمليات تكنولوجيا المعلومات (IT)، وعمليات سوق تكنولوجيا المعلومات. حيث يتم استخدام مجموعة متنوعة من الأجهزة في تقنية الحوسبة السحابية بما فيها الحواسب المكتبية، الحواسب المحمولة، الهواتف الذكية والأجهزة اللوحية وأجهزة المساعدة الرقمية للوصول إلى البرامج، ومساحات التخزين، ومنصات تطوير التطبيقات عبر الإنترنت، عن طريق خدمات مقدمة من قبل مزودي الحوسبة السحابية. وتشمل مزايا تقنية الحوسبة السحابية التوفير في التكاليف، ونسبة التوافر العالية، وسهولة الاستيعاب.

يُظهر الشكل (2-1) ست مراحل من نماذج الحوسبة، بدءاً من (محطة–جهاز رئيسي)، وصولاً إلى الحواسب المكتبية، حوسبة الشبكات، وحتى الحوسبة السحابية والشبكية.

في المرحلة الأولى، يتقاسم العديد من المستخدمين مراكز كبيرة قوية باستخدام محطات وهمية. وفي المرحلة الثانية، تُصبح الحواسب المكتبية القائمة بذاتها قوية بما فيه الكفاية لتلبية معظم احتياجات المستخدمين. أما المرحلة الثالثة، فيتم فيها توصيل الحواسب المكتبية، والحواسب المحمولة، والمخدمات مع بعضها البعض من خلال الشبكات المحلية لمشاركة الموارد وزيادة الأداء. وفي المرحلة الرابعة، تم ربط الشبكات المحلية بالشبكات المحلية الأخرى لتشكيل شبكة عالمية مثل الإنترنت للاستفادة من التطبيقات عن بعد والموارد الأخرى. وفي المرحلة الخامسة، وفرت شبكة

الحوسبة قوة حوسبة مشتركة ومساحة تخزين من خلال نظام الحوسبة الموزع. أما المرحلة السادسة، فقد وفرت الحوسبة السحابية المزيد من الموارد المشتركة على الإنترنت بطريقة متدرجة وبسيطة[5].

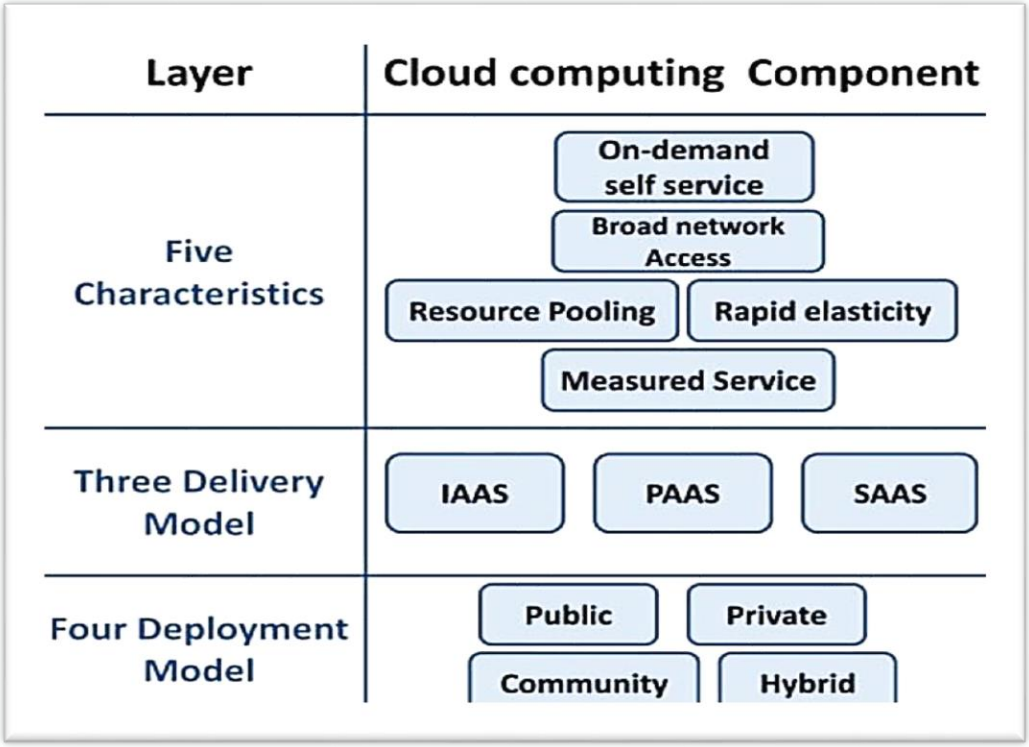


الشكل (1-2) ست نماذج حوسبة

2-3-تعريف الحوسبة السحابية[6]

يوجد العديد من التعاريف للحوسبة السحابية فكل شركة أو مؤسسة أو فرد يعرّف الحوسبة السحابية بطريقته ولكن التعريف الأشمل هو تعريف (the National Institute of Standards and Technology [NIST]) المعهد الوطني للمعايير والتكنولوجيا حيث ذُكر في هذا التعريف أن الحوسبة السحابية هي نموذج متكامل لتمكين الوصول المناسب عند الطلب إلى موارد الحوسبة القابلة للتشكيل (المعالجة، المخدمات، الشبكات، سعات التخزين والتطبيقات) عبر الشبكة والتي يمكن توسيعها أو تقليصها

بسهولة وبأقل جهد في الإدارة. ويشمل هذا التعريف على خمس خصائص جوهرية للحوسبة السحابية وثلاث نماذج خدمة وأربع نماذج نشر للسحابة كما هو مبين في الشكل (2-2).



الشكل (2-2) نموذج (NIST) للحوسبة السحابية

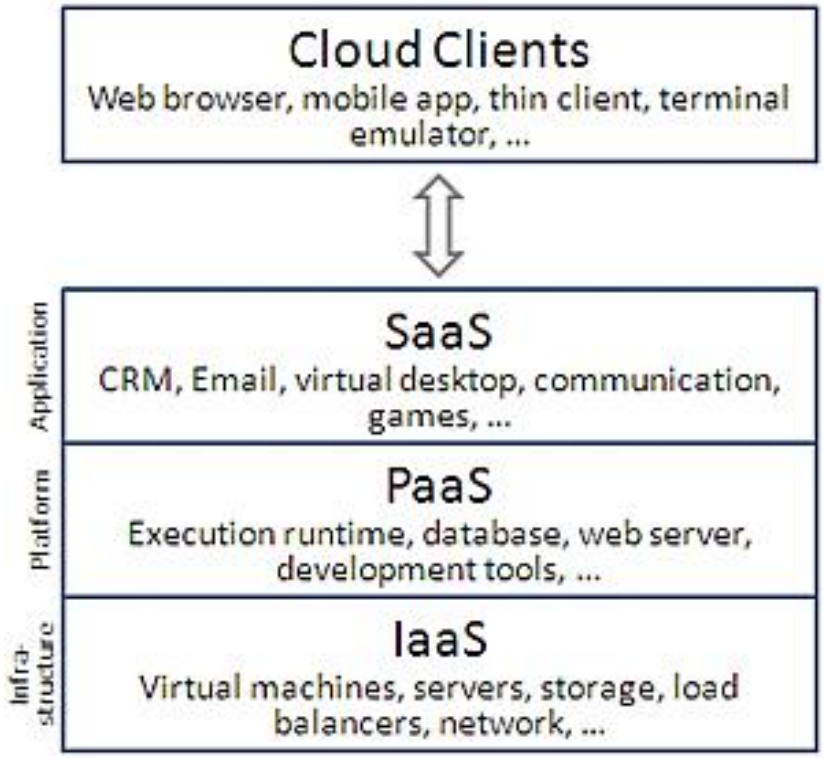
وهنا تجدر الإشارة إلى تعريف مصطلح البنية التحتية كما ورد في نموذج تعريف الحوسبة السحابية المنشور من قبل المعهد الوطني للمعايير والتكنولوجيا، حيث يقول التعريف إن البنية التحتية للسحابة هي مجموعة من العتاد الفيزيائي والبرمجي الذي يؤمن الخصائص الجوهرية الخمسة للحوسبة السحابية ومن منظور آخر تحتوي البنية التحتية للحوسبة السحابية على طبقتين طبقة فيزيائية وطبقة التجريد، الطبقة الفيزيائية تتألف من موارد العتاد الضرورية لإيصال خدمات السحابة، وتتضمن المخدمات والتخزين ومكونات الشبكة، أما طبقة التجريد تتألف من البرمجيات التي يتم نشرها على الطبقة الفيزيائية، ومن الناحية النظرية فان طبقة التجريد تتوضع فوق الطبقة الفيزيائية.

2-3-1-الخصائص الجوهرية للحوسبة السحابية

1. الإمداد الذاتي بناءً على الطلب (On-demand Self Service): في الحوسبة السحابية يكون المستخدم قادراً على زيادة أو تقليص الموارد دون الحاجة إلى التدخل البشري من قبل مزود السحابة.
2. الوصول عبر الشبكة (Broad Network Access): جميع الموارد متاحة عبر الشبكة ويتم الوصول إليها من مختلف أنواع الأنظمة والأجهزة (هواتف ذكية، حواسيب محمولة، أجهزة لوحية) ومن أي مكان وفي أي وقت.
3. تجميع الموارد (Resource Pooling): يتم تجميع موارد الحوسبة لتخدم العديد من المستخدمين حيث يتم إدراج أو إلغاء إدراج الموارد (المعالجة، التخزين، الذواكر، عرض نطاق الشبكة) سواء أكانت فيزيائية أو افتراضية وذلك لتلبية متطلبات المستخدمين وليس من الضروري أن يكون لدى المستخدم أي معرفة مسبقة بطريقة إدارة أو التحكم بهذه الموارد أو حتى بمكان تواجد هذه الموارد.
4. المرونة السريعة (Rapid Elasticity): يتم زيادة موارد الحوسبة أو إنقاصها بناءً على طلب المستخدم بشكل تلقائي وبمرونة عالية دون التدخل البشري من قبل مزود السحابة بحيث يمكن أن تكون الموارد المتاحة غير محدودة ومتاحة بأي كمية وفي أي وقت.
5. الخدمة المُقاسة (Measured Service): في الأنظمة السحابية يتم وبشكل تلقائي التحكم في استخدام الموارد وتحسينها بالاستفادة من قدرات القياس بناء على نوع الخدمة، حيث تتم مراقبة الموارد المستخدمة وتزويد كل من مزود السحابة والمستخدم بتقارير كاملة عنها.

2-3-2-نماذج الخدمة

الشكل (2-3) يوضح نماذج الخدمة التي يوفرها مزود السحابة.



الشكل (3-2) نماذج الخدمة التي يوفرها مزود السحابة

1. البرمجيات كخدمة (Software as a Service [SaaS]): يتم استخدام قدرة الحوسبة المقدمة إلى المستخدم لتشغيل التطبيقات على البنية التحتية للسحابة. حيث تكون التطبيقات متاحة للوصول من مختلف أنواع أجهزة المستخدم عن طريق مستعرض الويب أو واجهة البرنامج. وفي هذا النموذج لا يدير المستخدم ولا يتحكم بالطبقات التحتية للسحابة والتي تتضمن الشبكة والمخدمات وأنظمة التشغيل والتخزين أو حتى قدرات التطبيق الذي يستخدمه وإعداداته باستثناء بعض الصلاحيات التي تُمنح لبعض المستخدمين.
2. المنصة كخدمة (Platform as a Service [PaaS]): يتم استخدام قدرة الحوسبة المقدمة من السحابة إلى المستخدم لنشر وإنشاء تطبيقاته باستخدام لغات البرمجة والمكتبات والأدوات والخدمات التي يوفرها مزود السحابة وأيضا في هذا النموذج لا يدير المستخدم ولا يتحكم بالطبقات التحتية من بنية السحابة والتي تتضمن الشبكة والمخدمات وأنظمة التشغيل وسعات التخزين ولكنه يتحكم بالتطبيقات التي ينشرها وينشئها وأيضاً لديه صلاحيات لتعديل إعدادات بيئة المضيف للتطبيق (Application Hosting Environment).

3. البنية التحتية كخدمة (Infrastructure as a Service [IaaS]): في هذا النموذج يتم استخدام قدرة الحوسبة المقدمة إلى المستخدم لتوفير المعالجة والتخزين والشبكة والموارد الأساسية الأخرى حيث يستطيع المستخدم تشغيل البرمجيات بشكل اختياري لا على التعيين والتي تتضمن أنظمة التشغيل والتطبيقات. في هذا النموذج لا يدير المستخدم ولا يتحكم بالطبقات التحتية لبنية السحابة ولكن يتحكم بأنظمة التشغيل بصلاحيات كاملة وبسعات التخزين التي يوفرها مزود السحابة وأيضاً يتحكم بتطبيقاته الخاصة مع بعض الصلاحيات المحدودة للتحكم ببعض مكونات الشبكة مثل (Host Firewall).

2-3-3- نماذج نشر السحابة

1. السحابة الخاصة (Private Cloud): هنا يتم توفير البنية التحتية للسحابة للاستخدام بشكل حصري من قبل منظمة واحدة تشمل عدة مستخدمين ويمكن أن يتم إدارة وتشغيل وامتلاك السحابة من قبل هذه المنظمة أو طرف آخر أو بشكل مشترك بين المنظمة وطرف آخر كما ويمكن لهذه البنية التحتية أن تتواجد داخل أو خارج مبنى المنظمة.
2. السحابة التجميعية (Community Cloud): هنا يتم توفير البنية التحتية للسحابة للاستخدام بشكل حصري من قبل مجموعة من المستخدمين التابعين لمنظمات أو مؤسسات ذات اهتمامات مشتركة (مهمة معينة، متطلبات أمن) ويمكن أن يتم إدارة وتشغيل وامتلاك السحابة من قبل منظمة من هذه المنظمات أو أكثر أو من قبل طرف آخر أو بشكل مشترك بين المنظمات وطرف آخر كما ويمكن أن تتواجد البنية التحتية داخل أو خارج مبنى المنظمات.
3. السحابة العامة (Public Cloud): هنا يتم توفير البنية التحتية للسحابة للاستخدام من قبل العامة ويمكن أن يتم إدارة وتشغيل وامتلاك السحابة من قبل منظمة أكاديمية أو حكومية أو تجارية أو بالشراكة بين عدة منظمات وتتواجد البنية التحتية للسحابة في مبنى المنظمة.
4. السحابة الهجينة (Hybrid Cloud): تكون البنية التحتية لهذه السحابة عبارة عن تركيب لاثنين أو أكثر من البنى التحتية لسحابات مختلفة (خاصة، عامة، تجميعية) وكل منها تحافظ على كيانها الفريد ولكنها مرتبطة مع بعضها بتقنية موحدة بحيث تتيح إمكانية نقل التطبيقات والبيانات بين السحب مثال (إنشاء سحابة لموازنة الحمل بين السحب).

2-4-الحوسبة السحابية مقابل الخدمات السحابية

في هذا القسم، سوف نقدم جدولين يُظهران الفروق والسمات الرئيسية للحوسبة السحابية مقابل الخدمات السحابية[7].

الجدول -2-1-السمات الأساسية للحوسبة السحابية

السمات	الوصف
نظم البنى التحتية	تشمل المخدمات، وحدات التخزين، الشبكات التي يمكن توسيعها حسب طلب المستخدم
برامج التطبيقات	توفر واجهة مستخدم معتمدة على الويب، خدمات ويب لواجهات برمجة التطبيقات.
برنامج تطوير ونشر التطبيقات	يدعم تطوير وتكامل برنامج التطبيقات السحابية
برنامج إدارة النظام والتطبيقات	يدعم توفير الخدمات الذاتية السريعة والتكوين ومراقبة الاستخدام
شبكات IP	تربط المستخدمين النهائيين بالسحابة ومكونات البنية التحتية

الحوسبة السحابية هي أساس تكنولوجيا المعلومات للخدمات السحابية، كما أنها تتكون من تقنيات تتيح الخدمات السحابية. وتظهر السمات الأساسية للحوسبة السحابية في الجدول -2-1-.

الجدول -2-2-السمات الأساسية للخدمات السحابية

السمات	الوصف
الوصول إليها عبر الإنترنت	يتم الوصول إلى الخدمات عبر أساس معياري، وشبكة وصول عالمية.
المهارات التقنية المطلوبة	هناك مواصفات مُبسطة للمتطلبات
المتاحية	ويشمل طلب الخدمة الذاتية، النشر في الوقت الحقيقي تقريباً
السعر	يعتمد السعر على قدرات الاستخدام القائمة وجودتها
واجهة المستخدم	واجهة المستخدم تتضمن متصفحات لأجهزة متنوعة مع قدرات عالية
واجهة النظام	تعتمد واجهات النظام على خدمات Web الخاصة بواجهات برمجة التطبيقات لتوفير إطار معياري للوصول والدمج بين الخدمات السحابية
الموارد المشتركة	يجري مشاركة الموارد بين مستخدمي الخدمات السحابية، ومع ذلك، فإنه عبر خيارات تكوين الخدمة تتوفر إمكانية تخصيص الموارد.

2-5-مميزات الحوسبة السحابية

توفر الحوسبة السحابية عدد من المميزات الجديدة مقارنة بنماذج الحوسبة الأخرى [8]. وهذه المميزات يتم وصفها باختصار في هذا الجزء.

- **قابلية التطوير وتوفر الخدمات عند الطلب**

توفر الحوسبة السحابية الموارد والخدمات للمستخدمين عند الطلب. كما أن الموارد قابلة للتطوير عبر مراكز بيانات متعددة.

- **واجهة تعتمد على المستخدم**

واجهات السحابة تُعد مستقلة الموقع ويمكن الوصول إليها عن طريق واجهات ديناميكية مثل خدمات الويب ومتصفحات الإنترنت.

- **ضمان جودة الخدمة (Quality of Service [QoS])**

يمكن للحوسبة السحابية ضمان جودة الخدمة للمستخدمين من حيث أداء الأجهزة، وحدة المعالجة المركزية، معدل نقل البيانات وسعة الذواكر.

- **نظام التحكم الذاتي**

إن أنظمة الحوسبة السحابية هي أنظمة مستقلة تدار بشفافية للمستخدمين، فإن البرامج والبيانات الموجودة داخل السحابة يمكن إعادة تكوينها ودمجها تلقائياً تبعاً لاحتياجات المستخدم.

- **السعر**

استخدام الخدمات التي توفرها الحوسبة السحابية لا يتطلب تمويل كبير فلا حاجة لوجود نفقات رأسمالية حيث يمكن للمستخدمين دفع ثمن الخدمات والسعة حسب الحاجة والشعار المتبع من قبل مزودي الخدمات السحابية ادفع كما تستخدم (Pay as you Go)[9].

2-6-معايير الحوسبة السحابية

حتى الآن، لم يتم تطوير معايير الحوسبة السحابية بالكامل، ولكن هناك عدد من المعايير القائمة التي سهلت نمو هذا النوع من الحوسبة. الجدول-3-2- يوضح العديد من هذه المعايير المفتوحة والتي تستخدم حالياً في مجال الحوسبة السحابية[10].

الجدول –2-3– معايير الحوسبة السحابية

البرامج	الاتصالات	HTTP, XMPP
	الأمن	OAuth, OpenID, SSL/TLS
	التّعريب	Atom
العميل	المتصفحات	AJAX
	العمل بدون إنترنت	HTML5
تطبيقات	المحاكاة الافتراضية	OVF
المنصة	حزمة الحل	LAMP
الخدمة	البيانات	XML, JSON
	خدمات الويب	REST

2–7–منصات الحوسبة السحابية

في ظل الركود الاقتصادي العالمي الذي حصل مؤخراً توجهت شركات تكنولوجيا المعلومات العالمية لتحسين استخدام واستثمار تكنولوجيا المعلومات من ناحية العتاد الفيزيائي والبرمجي ونهضت بثورة في هذا المجال تم تسميتها الحوسبة السحابية وكان الغرض منها تخفيض رأس المال المستخدم والمصروف على تكنولوجيا المعلومات والاستثمار الأمثل لكافة موارد الحوسبة المستخدمة والتوافق مع متطلبات وشروط السوق بمرونة عالية. في عام 2007 كانت نتائج البحث عن مصطلح (Cloud Computing) في محرك بحث جوجل تعطي تقريباً مليوني نتيجة بينما في عام 2009 أعطت أكثر من 12 مليون نتيجة واليوم في عام 2016 أصبح البحث عن هذا المصطلح يعطي عدد كبير جداً من النتائج يصل إلى 193 مليون نتيجة وهذا الازدياد الهائل يشير إلى مدى الاهتمام الكبير الذي يبديه العالم بهذه التقنية، قدمت شركة (Saleforce) تطبيقاً كخدمة وطرحت شركة (Amazon) البنية التحتية كخدمة وبعد ذلك تم طرح المنصة كخدمة من قبل شركة (Microsoft) والكثير من الشركات العاملة في هذا المجال، ويعرض الجدول –2-4– مزودي منصات الحوسبة السحابية وعروضهم الرئيسية[11].

الجدول –2-4–مزودي منصات الحوسبة السحابية

الشركة	منصة الحوسبة السحابية	السنة	العروض الرئيسية
Amazon. com	خدمات أمازون للويب (Amazon Web Service)	2006	البنية التحتية كخدمة (التخزين، المعالجة، قواعد البيانات)
Google	محرك تطبيقات جوجل	2008	منصة تطبيقات الويب كخدمة
IBM	Blue Cloud	2008	مركز بيانات(Blue Cloud) الافتراضي
Salesforce.com	Force.com	2008	إطار تطبيقات الويب كمنصة عند الطلب
Microsoft	Azure	2009	منصة عمل التطبيقات كخدمة

2–8–التسعير

إن تحديد أسعار خدمات السحابة تعتمد على ثلاثة أبعاد رئيسية هي: سعة التخزين، معدل نقل البيانات، قدرة الحوسبة.

- تُقاس سعة التخزين عادة كمتوسط كمية البيانات المُخزنة يومياً بالجيجا بايت على مدى فترة شهر.
- أما مُعدل نقل البيانات، فيُقاس عن طريق حساب كمية البيانات الكلية المنقولة داخل وخارج المنصة أثناء عمليات المعالجة. وبشكل عام، يعتبر نقل البيانات بين الخدمات داخل نفس المنصة مجاني في العديد من المنصات.
- وتُقاس قدرة الحوسبة كوحدات الوقت اللازمة لتشغيل نموذج، أو تطبيق، أو آلة لتقديم الخدمات. الجدول –2-5–، يُقارن سعر خدمتين من الخدمات المقدمة من المنصات الثلاثة الرئيسية للحوسبة السحابية[12].

الجدول –2-5–مقارنة بين أسعار بعض خدمات الحوسبة السحابية المقدمة من الشركات الرئيسية

الخدمة	الوحدة	Amazon	Google	Microsoft
البيانات المُخزنة	1جيجا بايت في الشهر	$0.10	$0.15	$0.15
وقت الحوسبة	ساعات فورية	$0.10 - $1.20	$0.10	$0.12

باختصار، من خلال تحليل تكاليف الحوسبة السحابية اعتماداً على خصائص التطبيق، فإن تكاليف نشر تطبيق ما، ربما تختلف وفقاً للمنصة المُختارة. ومن الجدول –2-5–، يبدو أن سعر الوحدة

للمنصات الثلاثة الرئيسية مماثلة إلى حد كبير. وبجانب سعر الوحدة، من المهم ترجمتها إلى تكاليف تطوير، ونشر، وصيانة شهرية للتطبيق.

2-9-المكونات الرئيسية للسحابة

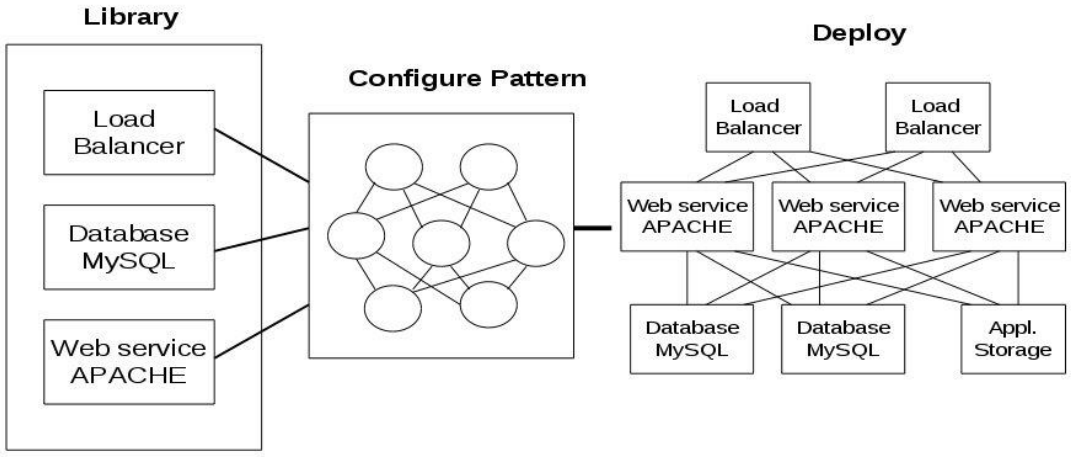
إن المكونات الرئيسة لمنصات الحوسبة السحابية تتضمن مخدّمات، ومساحات تخزين، وبنية تحتية، وبرمجيات، وأنظمة تشغيل، ومنصات عمل افتراضية. أما بالنسبة لأفضل الشركات المزودة لمكونات الحوسبة السحابية، فتُعرض في الجدول -2-6-.

الجدول -2-6-الشركات الرائدة في مجال مكونات الحوسبة السحابية

مكونات الحوسبة السحابية	الشركات
أجهزة الكمبيوتر	Dell, HP, IBM, Sun
مساحة التخزين	Sun, EMC, IBM
البنية التحتية	Cisco, Juniper Networks, Brocade Communication
البرمجيات	Eucalyptus. G-Eclipse. Hadoop
أنظمة التشغيل	Solaris, AIX, Linux (Red Hat, Ubuntu)
المنصات الافتراضية	Citrix, VMWare, IBM, Xen, Linux KVM, Microsoft

2-10-مثال على نشر تطبيق ويب

في هذا القسم، سوف نقدم مثال يبين كيفية الجمع بين المحاكاة الافتراضية والخدمة الذاتية لتسهيل نشر التطبيق[13]. في هذا المثال، نستخدم نشر تطبيق ويب باستخدام السحابة، كما هو موضح في الشكل -2-4-.



الشكل (2-4) مثال على نشر تطبيق

الخطوات التالية تمثل عملية نشر التطبيق:

- يختار المطور موازن الحمل، ومخدم الويب، وتطبيقات مخدم قاعدة البيانات من مكتبة صور الالة الافتراضي المُعدة مسبقاً.
- يقوم المطور بتكوين كل مكون لإعداد صورة مخصصة. حيث يتم تكوين موازن الحمل، وملء مخدم الويب بمحتواه الثابت عن طريق رفعه إلى سحابة التخزين، وتعبئة أدوات مخدم قاعدة البيانات بالمحتوى الديناميكي للموقع.
- ثم يقوم المطور بعد ذلك بتركيب الكود المخصص في البنية الجديدة، وبهذه الطريقة يجعل المكونات تتلاءم مع متطلبات تطبيق معين.
- يختار المطور نمط يأخذ صور كل طبقة ويقوم بنشرها، والتعامل مع الشبكات، والأمن، ومهام التطور.

إن تطبيق الويب الآمن، والمتوفر بكثرة موجود الآن ويعمل. وعندما يحتاج التطبيق لتحديث، يمكن تحديث صور الالة الافتراضية، ونسخها عبر سلسلة التطوير، ويمكن إعادة نشر البنية التحتية بأكملها.

في هذا المثال، يمكن استخدام مجموعة معيارية من المكونات لنشر تطبيق بسرعة. ومع هذا النموذج، يمكن تلبية احتياجات المشاريع التجارية بسرعة بدون الحاجة لاستهلاك الكثير من الوقت، وشراء دليل الاستخدام، والتركيب، والكابلات، وتكوين المخدمات، ووحدات التخزين، والبنية التحتية للشبكة. وقد كانت المشاريع الصغيرة والمتوسطة أول من تبنى الحوسبة السحابية. ومع ذلك، فقد ظهرت مؤخراً عدداً من الأمثلة لاستخدام الحوسبة السحابية في الشركات الكبيرة. ويوضح الجدول –7-2– ثلاثة أمثلة لاستخدام الحوسبة السحابية في الشركات الكبيرة.

الجدول –7-2–أمثلة الحوسبة السحابية في الشركات الكبيرة

الشركة	السيناريو	الاستخدام	الفوائد
Eli Lilly	حوسبة البحث والتطوير عالية الأداء	مخدم أمازون ووحدة التخزين العنقودية	وقت نشر سريع بتكلفة أقل
New York Times	تحويل البيانات	تحويل المواد الأرشيفية (3مليون) إلى صيغ بيانات جديدة باستخدام خدمة حوسبة أمازون المرنة	توفير سريع ومرونة عالية على موارد البنية التحتية
Pitney Bowes	تطبيقات (B2B)	استضافة تطبيق نموذج الطباعة الإلكتروني للعملاء يستخدم خدمات MS-SQL, Azure.net	مرونة عالية بتكلفة أقل

2-11-تحديات الحوسبة السحابية

باختصار، النموذج الجديد للحوسبة السحابية يوفر عدداً من الفوائد والمزايا تتفوق على نماذج الحوسبة السابقة وهناك العديد من المنظمات التي تعتمده وتتبناه. ومع ذلك، لا يزال هناك عدد من التحديات، والتي تُعالج حالياً من قبل الباحثين والممارسين على أرض الواقع. وفيما يلي عرض لهذه التحديات بإيجاز [7].

2-11-1-الأداء

المشكلة الكبرى في الأداء ربما تكون بالنسبة لبعض التطبيقات الموجهة نحو المعاملات وغيرها من تطبيقات البيانات الكثيفة، وفيها، قد تفتقر الحوسبة السحابية للأداء المناسب. كما أن المستخدمين البعيدين عن مزودي السحابة ربما يواجهون تأخير وخمول عالي.

2-11-2-الأمن والخصوصية

لا تزال الشركات تشعر بالقلق إزاء الأمن عند استخدام الحوسبة السحابية. فالعملاء قلقون بخصوص التعرض لهجمات عندما تكون المعلومات الهامة ومصادر تقنية المعلومات خارج جدار الحماية. وتفترض حل مشكلة الأمن على مقدمي الحوسبة السحابية اتباع سياسات الأمان المعيارية.

2-11-3-التحكم

تشعر بعض أقسام تكنولوجيا المعلومات بالقلق لأن مقدمي الحوسبة السحابية يمتلكون السيطرة الكاملة على المنصات. كما أن مزودي الحوسبة السحابية لا يقومون عادة بتصميم منصات لشركات محددة وممارساتها التجارية.

2-11-4-تكاليف معدل نقل البيانات

يمكن للشركات توفير الأموال المهدورة على الأجهزة والبرمجيات باستخدام الحوسبة السحابية، إلا أنها يمكن أن تتحمل ارتفاع رسوم معدل نقل البيانات للشبكة. وقد تكون تكلفة معدل نقل البيانات منخفضة بالنسبة لتطبيقات الإنترنت الصغيرة، والتي لا تُعد كثيفة البيانات، ولكنها يمكن أن ترتفع بشكل كبير بالنسبة لتطبيقات البيانات الكثيفة.

2-11-5-الدقة والموثوقية

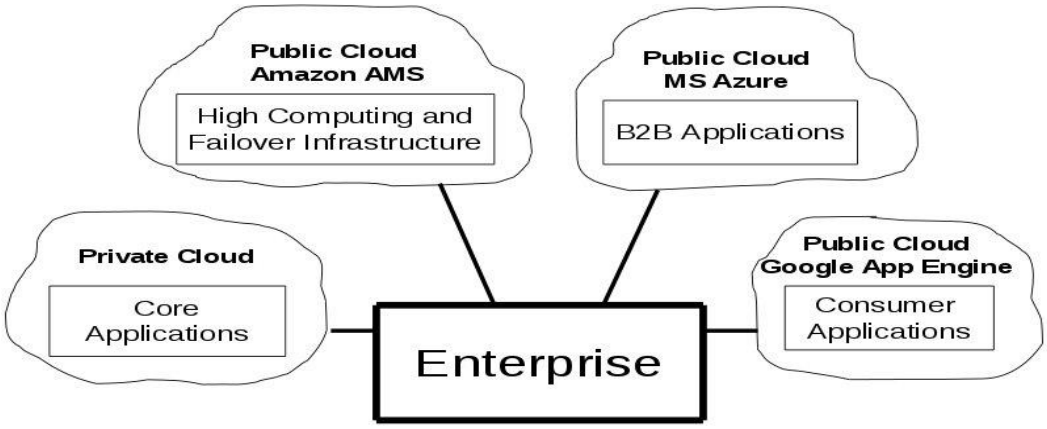
لا تقدم الحوسبة السحابية موثوقية دائمة على مدار الساعة. حيث كانت هناك بعض الحالات التي تعاني فيها خدمات الحوسبة السحابية من انقطاع لعدد قليل من الساعات. وفي المستقبل، نتوقع أن نرى المزيد من مزودي الحوسبة السحابية. وفي ساحة الأبحاث، أطلقت معامل (HP, Intel, Yahoo) اختبارات (Cloud Research Test Bed) بعدة أماكن في آسيا، أوروبا، وأمريكا الشمالية

بهدف تطوير الابتكارات وزيادة الموثوقية. كما أطلقت (IBM) بحوث الحوسبة السحابية والتي تعتبر بمثابة مجموعة من موارد الحوسبة المتاحة للوصول عالمياً حسب الطلب والتي تدعم العمليات التجارية [7].

2-12-الحوسبة السحابية في المستقبل

باختصار، إن الحوسبة السحابية هي بالتأكيد نموذج من نماذج الحوسبة التي ستبقى متوفرة لفترة طويلة. وفي المستقبل القريب، يمكن أن تنبثق الحوسبة السحابية في اتجاهات مختلفة. وأحد السيناريوهات المحتملة للمستقبل هو أن المؤسسة ربما تستخدم سحابة هجينة موزعة كما هو موضح في الشكل -2-5-.

ووفقاً لهذا السيناريو، فإن المؤسسة سوف تستخدم التطبيقات الأساسية على سحابتها الخاصة، في حين سيتم توزيع بعض التطبيقات الأخرى على عدة سحب خاصة والتي عُدلت من أجل تطبيقات محددة[14].



الشكل (2-5) بنية سحابة هجينة موزعة

2-13-الخاتمة

قدمنا في هذا الفصل تعريف الحوسبة السحابية وأنواعها، وأهم ميزاتها وخصائصها، والتحديات المستقبلية التي تواجهها.

الفصل الثالث

موازنة الحمل في الحوسبة السحابية

الفصل الثالث

موازنة الحمل في الحوسبة السحابية

3-1-مقدمة

في الحوسبة السحابية يتم تخزين البيانات والموارد الموزعة في بيئة مفتوحة، وكمية البيانات المخزنة تزداد بشكل سريع في البيئات المفتوحة لذلك فان مسألة موازنة الحمل واحدة من المشاكل الأساسية في بيئة السحابة. يوجد العديد من الخوارزميات التي تُقدم موازنة حمل فعالة واستثمار أفضل لموارد النظام. وفي الحوسبة السحابية يوجد العديد من أنواع الحمل التي يمكن تظهر ومنها الحمل في الذواكر والحمل في المعالجة والحمل في الشبكة، وموازنة الحمل هي العملية التي يتم فيها إيجاد العقد ذات الحمل العالي ومن ثم نقل الحمل الزائد إلى عقد أخرى.

3-2-موازنة الحمل

موازنة الحمّل هي عملية توزيع الأحمال الكبيرة على عقد المعالجة الصغيرة للحصول على أفضل زمن استجابة والاستثمار الأمثل للموارد الفيزيائية والبرمجية وبالتالي تحسين الأداء الكلي للنظام[15] [16]. تساعد موازنة الحمل على التوزيع العادل لموارد الحوسبة وذلك بهدف تحقيق مستوى عالي من رضا المستخدم. بالتالي فإن الاستخدام الصحيح للموارد وموازنة الحمل السليمة ستؤدي بدورها إلى تقليص كمية الموارد المستخدمة وتساعد في تجنب الفشل وتجنب حالات عنق الزجاجة (Bottlenecks).

وتقسم خوارزميات موازنة الحمل إلى نهجين:

1) النهج الستاتيكي (Static Approach): إن هذا النهج لا يعتمد على حالة النظام الراهنة ويوجد علم مسبق بحاجات النظام ويتم تعريف هذا النهج أثناء تصميم وتطبيق النظام.
2) النهج الديناميكي (Dynamic Approach): في هذا النهج تستند قرارات موازنة الحمل فقط على الحالة الراهنة للنظام. لا يوجد حاجة لمعرفة مسبقة. لذلك هو أفضل من النهج الستاتيكي بالنسبة للأنظمة الموزعة مثل الحوسبة السحابية.

وتستخدم تقنية موازنة الحمل لتوزيع الحمل بشكل موحد بين جميع نقاط الشبكة بأقصى قدر من معدل التدفق (Throughput) وزمن الاستجابة (Respond Time) وبالتأكيد مع تجنب الحمل الزائد بين عقد الشبكة[17]. ومن الضروري تنفيذ موازنة الحمل بشكل صحيح لأن الفشل في عقدة من العقد يمكن أن يؤدي إلى عدم توافر البيانات.

يجب أن تتجنب خوارزمية موازنة الحمل المثالية الحمل الزائد أو تحت الحمل على عقدة معينة. ومهمة اختيار خوارزمية موازنة الحمل في بيئة الحوسبة السحابية ليست سهلة بسبب وجود قيود إضافية مثل الأمن والموثوقية ومعدل التدفق وما إلى ذلك. يمكن لخوارزميات موازنة الحمل أن تعمل بنمطين: تعاونية وغير تعاونية، في التعاونية تعمل العقد في وقت واحد من أجل تحقيق الهدف المشترك المتمثل في تحسين زمن الاستجابة الكلي، أما في النمط الغير تعاونية يتم تشغيل المهام بشكل مستقل من أجل تحسين زمن الاستجابة للمهام المحلية. وهنا تجدر الإشارة إلى أن خوارزمية موازنة الحمل الستاتيكية لا تأخذ بعين الاعتبار حالة وسلوك العقدة عند توزيع الحمل، بينما تتحقق خوارزمية الموازنة الديناميكية مسبقاً من حالة العقدة حين توزيع الحمل.

3-3-مقاييس تقنيات موازنة الحمل

لابدّ أن يكون هناك بعض المقاييس لتقييم أداء تقنيات موازنة الحمل لتحسين توزيع الموارد الفيزيائية والبرمجية عند تلقي طلبات المستخدم. نورد في هذه الأطروحة بعض المقاييس لتقييم تقنيات موازنة الحمل:

1. معدل التدفق (Throughput): يتم استخدام معدل التدفق لحساب عدد المهام المكتملة التنفيذ، يزداد أداء النظام بزيادة معدل التدفق وبمعنى أخر هي كمية الخرج التي يمكن للنظام ينفذها بالنسبة لكمية الدخل الوارد في فترة معينة من الزمن.
2. التسامح بالأخطاء (Fault Tolerance): تعني الاستعادة من الأعطال التي قد يتعرض لها النظام، تقنية موازنة الحمل يجب أن تحقق الاستعادة من الأعطال بشكل جيد.
3. زمن التهجير (migration time): هو الزمن المطلوب لتهجير المهام من عقدة إلى أخرى وكلما كان هذا الزمن أقل كلما كان أداء النظام أفضل.
4. زمن الاستجابة (response time): هو الزمن المطلوب لوضع طلب المستخدم في مرحلة التنفيذ أي هو الزمن الذي تستغرقه خوارزمية الموازنة للاستجابة على الطلب وكلما كانت قيمة هذا البارامتر أقل كلما كان أداء النظام أفضل.
5. التوسعيّة (Scalability): هي قدرة خوارزمية موازنة الحمل على تحقيق الموازنة بأي عدد منتهي من عقد النظام وحتى تكون الخوارزمية جيدة يجب تحقيق هذا البارامتر.

3-4-سياسة خوارزميات موازنة الحمل[18][19]

هناك العديد من السياسات المستخدمة لضبط خوارزميات موازنة الحمل نورد منها:

1. سياسة المعلومات (Information Policy): هذه السياسة تعرّف ماهية المعلومات المطلوبة وكيفية جمع هذه المعلومات وأيضاً متى يتم جمع هذه المعلومات.
2. سياسة التشغيل (Triggering Policy): الفترة الزمنية لبدء تشغيل عمليات موازنة الحمل والبدء بإدارة الحمل.
3. سياسة نوع الموارد (Resource Type Policy): في هذه السياسة يتم تعريف كافة أنواع الموارد المتاحة خلال عملية موازنة الحمل.
4. سياسة الموقع (Location Policy): تُستخدم كل نتائج سياسة نوع الموارد لإيجاد شريك المخدّم أو المستقبل.
5. سياسة التحديد (Selection Policy): تُستخدم لاكتشاف المهمة الواجب نقلها من عقدة ذات حمل عالي إلى عقدة حرة أو خفيفة الحمل.

3-5- الأهداف الرئيسية لخوارزمية موازنة الحمل

1. الفعالية من حيث التكلفة: موازنة الحمل تساعد على تحسين أداء النظام بأقل كلفة ممكنة.
2. قابلية التوسع والمرونة: يمكن أن يتغير حجم النظام بعد فترة من تطبيق خوارزمية موازنة الحمل لذلك يجب على الخوارزمية المطبقة معالجة كافة الحالات والتغيرات وبالتالي يجب أن تكون هذه الخوارزمية مرنة وقابلة للتوسع.
3. الأولوية: يجب أن تحقق خوارزمية موازنة الحمل تحديد الأولويات بالنسبة للموارد والمهام وبالتالي المهام ذات الأولوية الأعلى تحصل على فرص تنفيذ أكثر.

3-6-خوارزميات موازنة الحمل الستاتيكية المستخدمة حالياً

3-6-1-موازنة الحمل الدّوارة (Round-Robin Load Balancer)

موازنة الحمل الدّوارة هي خوارزمية موازنة ستاتيكية لا تأخذ بعين الاعتبار حالة الحمل على العقدة عند تخصيص الوظائف. وهي تستخدم خوارزمية الجدولة الدّوارة لتخصيص الوظائف. حيث تقوم بتحديد عقدة بشكل عشوائي ثم توزع الوظائف على العقد بإسلوب دوّار (Round Robin Manner). وفي هذه الخوارزمية زمن تنفيذ أي عملية غير معروف قبل التنفيذ. وهناك احتمال بأن تحصل عقدة ما على حمل ثقيل، وبالتالي أُقترحت الخوارزمية الدوّارة المثقّلة (Weighted Round-Robin) لحل هذه المشكلة[20] [21]، حيث يتم تعيين وزن (ثقل) لكل عقدة و اعتماداً على وزن العقدة ستحصل على العدد المناسب من الطلبات. إن تعيين الوزن للعقد يفيد بالتحكم بحركة مرور البيانات على الشبكة.

3-6-2- MIN-MIN [22]

وهي خوارزمية موازنة حمل ستاتيكية. وهنا تجدر الإشارة إلى بعض المصطلحات المتعلقة بخوارزميات موازنة الحمل الستاتيكية:

1. زمن الحساب المتوقع (Expected Time to Compute [ETC]): يتم تخزين زمن التنفيذ المتوقع للمهام على كافة العقد في مصفوفة (ETC). إذا كانت المهمة غير قابلة للتنفيذ على عقدة معينة، يتم تعيين المدخل الخاص بهذه المهمة في مصفوفة (ETC) قيمة لانهاية (Infinity).
2. موازنة الحمل النفعية (opportunistic Load Balancing [OLB]): نُسند الأعمال إلى العقد بترتيب اعتباطي (لا على التعيين) متجاهلة الزمن المتوقع للحساب وتُقدم هذه الآلية جدولة موازنة حمل ولكن نتائجها سيئة من حيث زمن الانتهاء لتنفيذ المهام الموكلة للعقدة.
3. خوارزمية الحد الأدنى لزمن التنفيذ (Minimum Execution Time [MET]): كل مهمة تُسند إلى العقدة المذكور أنها صاحبة أدني زمن تنفيذ المهمة في مصفوفة (ETC) وذلك بغض النظر عن الحمل الحالي للعقدة وتحاول هذه الخوارزمية إيجاد أفضل زوج مهمة–عقدة ويتحسن فيها زمن انتهاء تنفيذ المهام على العقد ولكنها تسبب خللاً في توازن الحمل لتجاهلها الحمل الحالي للعقدة.
4. خوارزمية الحد الأدنى لزمن الاستكمال (Minimum Completion Time [MCT]): تعتمد هذه الخوارزمية عند إسناد المهام إلى العقد على أدنى زمن لاستكمال المهمة. يحسب زمن استكمال المهمة بجمع زمن تنفيذ المهمة على العقدة مع الزمن المطلوب لتكون العقدة جاهزة للبدء بتنفيذ المهمة (زمن استعداد العقدة). يتم اختيار العقدة صاحبة أدني زمن استكمال وتُسند إليها المهمة هذه الخوارزمية تدرس حالة إسناد المهمة إلى العقدة لمرة واحدة فقط.

خوارزمية (Min-Min) تبدأ بمجموعة المهام غير المُسندة إلى عقد، أولاً يتم حساب الحد الأدنى لزمن الاستكمال (MCT) لكافة المهام ثم يتم اختيار المهمة صاحبة أدنى زمن استكمال (Min(MCT)) وإسنادها إلى العقدة صاحبة أدنى زمن استكمال لجميع المهام ثم يتم تحديث جدول زمن استعداد العقد وتتكرر هذه العملية حتى يتم إسناد كافة المهام غير المسندة.

3-6-3- MAX-MIN [23] [24]

خوارزمية (Max-Min) هي تقريباً نفس الخوارزمية (Min-Min) باستثناء ما يلي: بعد معرفة الحد الأدنى لزمن الاستكمال للمهام (MCT)، يتم اختيار المهمة صاحبة اعلى زمن استكمال وإسنادها

إلى العقدة صاحبة الحد الأدنى لزمن استكمال جميع المهام. ثم يتم تحديث الزمن لاستعداد العقدة بإضافة زمن تنفيذ المهمة الموكلة.

3-6-4-موازن الحمل (Min-Min) [25]

خوارزمية LBMM (Load Balancer Min-Min): الهدف الرئيسي من هذه الخوارزمية هو تقليل زمن استكمال المهام الموكلة إلى العقدة (Make-Span) وتقوم هذه الخوارزمية بالخطوات التالية:

1. تنفيذ خوارزمية الجدولة (Min-Min) وحساب قيمة (Make-Span).
2. اختيار العقدة صاحبة اعلى قيمة (Make-Span).
3. بمقابل العقدة التي تم اختبارها تُحدد المهمة صاحبة أقل زمن تنفيذ.
4. تقارن أعلى زمن استكمال للمهمة المحددة مع قيمة (Make-Span) فإذا كان أقل منه يتم إسناد المهمة إلى العقدة صاحبة أعلى زمن استكمال، ويتم تكرار هذه الخطوات.
5. تتوقف هذه العملية عند إسناد كافة المهام.

إن أداء هذه الخوارزمية يكون أفضل كلما كان عدد المهام الصغيرة أكبر من عدد المهام الكبيرة.

3-6-5-خوارزمية موازنة الحمل Max-Min-Max [26]

يتم الجمع في هذه الخوارزمية بين خوارزميتي الجدولة (OLB) و (LBMM) حيث تكون مهمة (OLB) إبقاء كافة العقد في حالة عمل مستمر لغاية الوصول موازنة الحمل المرجوة أما خوارزمية (LBMM) تُستخدم لتقليل زمن تنفيذ كل مهمة على العقدة وبالتالي تقليل (MCT). إن عملية الجمع بين هاتين الخوارزميتين أدت إلى تحسين استثمار الموارد وتقوم هذه الخوارزمية بتنفيذ الخطوات التالية:

1. حساب معدل زمن الاستكمال لكل عقدة.
2. اختيار المهمة صاحبة أعلى زمن استكمال.
3. اختيار عقدة غير مسندة تملك أدنى زمن استكمال والذي يجب أن يكون أقل من أعلى زمن استكمال للمهمة المحددة وترسل المهمة إلى العقدة المحددة.
4. في حال تم إسناد مهام لكافة العقد تتم إعادة التقييم بالأخذ بعين الاعتبار العقد المسند والغير مسند لها مهام. ويتم حساب (MCT) كالتالي:
   - (MCT) للعقدة المسند إليها المهمة هو مجموع (MCT) لكافة المهام المسندة إلى العقدة و (MCT) للمهمة الحالية.
   - (MCT) للعقدة المسندة هو (MCT) للمهمة الحالية.
5. يتم تكرار الخطوات من 2 إلى 4 حتى يتم تنفيذ كافة المهام.

3-7- خوارزميات موازنة الحمل الديناميكية المستخدمة حالياً

3-7-1-النشر المتساوي للتنفيذ الحالي (Equally Spread Current Execution)

هي خوارزمية موازنة ديناميكية، تتعامل مع المهام على أساس الأولوية. وتحدد الأولوية للمهام عن طريق التحقق من حجم المهمة. تقوم هذه الخوارزمية بتوزيع الحمل بشكل عشوائي بالتحقق أولاً من حجم المهمة ثم نقل الحمل إلى العقدة الأخف حمولة. ينشر موازن الحمل الحمولة على مختلف العقد. وتسمى أيضا تقنية الطيف المنتشر.

3-7-2-خوارزمية موازن الحمل المخنوق (Throttled Load Balancer)

في الحوسبة السحابية يوجد عدد كبير من الآلات الافتراضية (Virtual Machines) ويمكن تصنيف هذه الآلات الافتراضية على أساس نوع الطلبات التي يمكن لهذه الآلات الافتراضية التعامل معها. يطلب في هذه الخوارزمية المستخدم أولاً من موازن الحمل العثور على آلة افتراضية مناسبة لتنفيذ العملية المطلوبة[27]. عندما يرسل العميل الطلب يقوم موازن الحمل في البداية بالبحث عن المجموعة التي يمكنها التعامل مع هذا الطلب ثم يقوم بتخصيص المهمة إلى الآلة الافتراضية الأخف حمولة في تلك المجموعة.

3-7-3-الحشد الفعال (Active Clustering)

الحشد الفعال هي خوارزمية أساسها الحشد (العنقدة)، في هذه الخوارزمية يتم تمثيل كل حشد (Cluster) على أنه مجموعة. المبدأ الكامن وراء مفهوم الحشد الفعال هو تجميع العقد[28] المتشابهة مع بعضها في مجموعات ومن ثم العمل على هذه المجموعات. إن عملية إنشاء الحشد تدور حول مفهوم العقدة صانعة الوفاق (Matchmaker Node)، في هذه العملية تختار العقدة الأولى عقدة جاره لها تسمى العقدة صانعة الوفاق والتي تكون من نوع مختلف. هذه العقدة (صانعة الوفاق) تؤمّن الاتصال مع جارتها التي هي من نفس النوع وتسمّى العقدة الأولية (Initial Node) ثم تنفصل العقدة صانعة الوفاق ويتم إعادة هذه العملية بشكل متكرر. تحسّن هذه الخوارزمية أداء النظام وتؤمن متاحيه عالية من الموارد، وبالتالي زيادة التدفق.

3-7-4-الانضمام إلى الصف الخامل (Join-Idle Queue)

تُستخدم خوارزمية الانضمام إلى الصف الخامل أساساً للأنظمة واسعة النطاق حيث تستخدم مُسندي المهام لموازنة الحمل بين المعالجات الخمولة عبر مُسندي المهام ثم تخصيص العمل

للمعالجات للحد من متوسط طول قائمة الانتظار في كل معالج. وعيب هذه الخوارزمية أنها غير قابلة للتوسيع. تم اقتراح هذه الخوارزمية لخدمات الويب القابلة للتطوير بشكل حيوي. حيث أنها تقلل بشكل فعّال من الحمل على النظام، ولا تسبب أية حمل على الاتصالات في وصول العمل ولا تؤثر على زمن الاستجابة الفعلية، وتقوم بأداء أقرب إلى المثالي في حال تم استخدامها في خدمات الويب[29].

3-8- الخاتمة

إنّ موازنة حمل النظام إحدى القضايا الحساسة. حيث تستخدم التقنيات والخوارزميات المختلفة لحلّ هذه المشكلة. في هذا الفصل قدمنا دراسة عن المقاييس المختلفة لتقييم أداء خوارزميات الموازنة بالإضافة إلى السياسات والأهداف المرجوة من هذه الخوارزميات بالإضافة إلى عرض دراسة عن طرائق موازنة الحمل المتنوعة في البيئات المختلفة.

وتم تصنيف الخوارزميات إلى ستاتيكية وديناميكية. وقمنا بعرض مختلف الخوارزميات التي يمكن تطبيقها في بيئات ستاتيكية. وبعدها عرضنا مختلف خوارزميات الموازنة الديناميكية. وبذلك نكون قد ناقشنا بعض الخوارزميات الإضافية التي يمكن أن تساعد في حل بعض المشاكل الفرعية في الموازنة التي تنطبق على الحوسبة السحابية.

الفصل الرابع

الحوسبة السحابية والآلات الافتراضية

الفصل الرابع

الحوسبة السحابية والآلات الافتراضية

4-1- تقنية الافتراضية (Virtualization)

في عالم الحوسبة، الافتراضية تشير إلى إنشاء إصدار افتراضي (يحاكي الإصدار الواقعي) من شيء ما، بما يتضمن منصات افتراضية للعتاد الصلب، أنظمة التشغيل، أجهزة التخزين وموارد الشبكات الحاسوبية. بدأت الافتراضية كطريقة للتقسيم المنطقي لموارد النظام التي توفرها منصات العمل الحاسوبية بين مختلف التطبيقات. منذ ذلك، انطلق مفهوم الافتراضية وبدأ بالتوسع[30].

4-2- تطبيق تقنية الافتراضية في العتاد الصلب

افتراضية العتاد الصلب هي تطبيق الافتراضية على الحواسب كمنصات عتاد صلب كاملة، سواء تطبيق المفهوم المجرد للافتراضية منطقياً على مكونات حاسوبية محددة كتقسيم القرص الصلب إلى سواقات، أو فقط من الناحية الوظيفية المطلوبة لتشغيل مختلف أنظمة التشغيل كما في حالة تشغيل الآلات الافتراضية. الافتراضية تخفي الخصائص الفيزيائية لمنصات الحوسبة عن المستخدمين، تقدم بالمقابل منصة حوسبة مجردة. يسمى التطبيق البرمجي الذي يتحكم بالتقنية الافتراضية "برنامج التحكم"، لكن مصطلح مراقب الآلة الافتراضية (Hypervisor) أصبحت لها الأفضلية مع مرور الوقت[31] [32]. تشير افتراضية العتاد الصلب أو افتراضية المنصة إلى عملية إنشاء آلة افتراضية والتي تعمل كحاسب حقيقي ذات نظام تشغيل حقيقي، هذه الآلة الافتراضية تحاكي الآلة الحقيقية بوظائفها. تتم العملية من خلال برنامج مراقب الآلة الافتراضية (Hypervisor)، يقوم بتشغيل البرامج الضيفة والتي هي في هذه الحالة آلات افتراضية تعمل عليها أنظمة تشغيل. النظام المستضيف يسمح بتنفيذ العديد من الآلات الافتراضية. هذه الآلة الافتراضية تعمل حينا وكأن لها وصول مباشر إلى الموارد الفيزيائية. عملية الوصول للموارد الفيزيائية للنظام (مثل الوصول الشبكي، شاشة العرض، لوحة المفاتيح وأقراص التخزين) تدار من مستوى مختلف عن معالج الحاسب المستضيف أو نظام ذاكرته. أيضاً الآلة الضيف لا تملك حق الوصول لجميع الأجهزة الملحقة للحاسب المستضيف، هناك سياسة وصول موضوعة تدير عملية الوصول للموارد الفيزيائية.

للتقنية الافتراضية تأثيرات سلبية من ناحية استهلاك الموارد المطلوبة لتشغيل برنامج مراقب الآلة الافتراضية (Hypervisor) إضافة لنظام تشغيلها ومن ناحية أخرى انخفاض الأداء المحتمل مقارنة عند تشغيل نفس التطبيق على آلة حقيقية.

4-2-1-أسباب تطبيق تقنية الافتراضية في العتاد الصلب

- في حالة دمج السيرفرات، العديد من السيرفرات الصغيرة يمكن استبدالها بسيرفر واحد كبير لزيادة استخدام الموارد المكلفة مثل وحدة المعالجة المركزية. على الرغم أن العتاد الصلب تم دمجه، فإن أنظمة التشغيل تبقى مستقلة. السيرفر الأكبر يقوم باستضافة عدة آلات افتراضية ضيفة. هذه ما يدعى بالانتقال فيزيائي-إلى-افتراضي (P2V).
- دمج السيرفرات له أيضاً فائدة كبيرة من ناحية تقليل استهلاك الطاقة.
- يمكن إدارة الآلة الافتراضية عن بعد بسهولة أكبر مقارنة بالفيزيائية. وإعداداتها تكون أكثر مرونة. هذا مفيد جدا في تطوير النواة وفي مناهج تعليم أنظمة التشغيل.
- يمكن تجهيز الآلة الافتراضية بمرونة بدون الحاجة لشراء عتاد صلب في كل مرة تحتاج فيها للتعديل.
- يمكن نقل الآلة الافتراضية من آلة فيزيائية إلى أخرى بسهولة، وفي حالة حدوث خطأ في الآلة الافتراضية فإنه لا يؤذي الآلة الفيزيائية المستضيفة أو نظام تشغيلها.

هناك العديد من المناهج والسيناريوهات لتطبيق الافتراضية على العتاد الصلب، مثلا:

- تشغيل تطبيق غير مدعوم من نظام التشغيل المستضيف: يتم تشغيل هذا التطبيق من خلال تجهيز آلة افتراضية داخل النظام المستضيف وتثبيت نظام تشغيل يدعم التطبيق المراد تشغيله.
- تقديم نظام تشغيل جديد: يعمل على الآلة الافتراضية مستقل تماماً عن النظام المستضيف.
- افتراضية السيرفر: بتشغيل عدة سيرفرات افتراضية على سيرفر فيزيائي واحد بما يتناسب مع الموارد الصلبة التي يمتلكها السيرفر الفيزيائي.
- إنشاء بيئة محمية: في حالة حدوث انهيار لنظام التشغيل الذي يعمل على الآلة الافتراضية وانعدام إمكانية تشغيله من جديد، يمكن إنشاء نسخة نظيفة خالية من الأخطاء وتشغيلها بأمان بدون حدوث أضرار على نظام التشغيل المستضيف.

4-2-2-أنماط تقنية الافتراضية في العتاد الصلب

1-الافتراضية الكاملة

يتم توفير عتاد صلب كاف يوفر تشغيل الآلة الافتراضية مع نظام تشغيلها الغير معدّل، هذا يعني أن نظام تشغيل هذه الآلة الافتراضية يتيح وصول كامل لكافة ميزات العتاد الصلب المتوفر كبنية تحتية، هنا جميع الآلات الافتراضية تملك نفس الإمكانيات بما يخص التعامل مع الآلة الفيزيائية كما هو الحال لنظام التشغيل التقليدي، بما يتضمن التحكم بأدوات الدخل والخرج، نظام المقاطعة، الوصول للذواكر، تعليمات لغة الآلة. في هذا النمط يكون أي برنامج قابل للعمل على آلة فيزيائية قابل للعمل على آلة افتراضية، وذلك على نفس نظام التشغيل المعتمد عليها، هنا تبقى المشكلة اختيار نظام التشغيل المناسب[33]. الأنماط الأخرى من الافتراضية تسمح فقط لنظام تشغيل معدّل ليعمل على الآلة الافتراضية. مثال مهم عن الافتراضية الكاملة هو برنامج التحكم لنظام تشغيل (CP/CMS) من (IBM)، والمعمول به منذ عام 1972 لليوم حيث يوفر لكل مستخدم حاسب افتراضي مستقل يحاكي الحاسب الفيزيائي. بالتالي هذا الحاسب الافتراضي والذي هو آلة افتراضية عملياً له مقدرات كاملة على البنية التحتية المتاحة. هذه المحاكاة تتيح سيطرة على كامل مقدرات الآلة الفيزيائية وينتج عنها آلة مفردة موزعة على عدة مستخدمين.

التحدي الرئيسي للافتراضية الكاملة هو التعامل مع محاكاة العمليات المسموحة مثل تعليمات الدخل/الخرج وعمليات القراءة والكتابة على المسجلات، أي أن أي عملية تنفذ من قبل آلة افتراضية يجب أن تبقى نتيجتها عند هذه الآلة ولا تؤثر على غيرها.

أن هذا النمط ينجح تماماً مع:

- مشاركة نظام التشغيل لحاسب على عدة مستخدمين.
- عزل المستخدمين عن بعضهم.
- محاكاة عتاد صلب جديد لإنجاز إنتاجية وموثوقية وأمان أعلى.

2-إفتراضية جزئية

هنا تتم محاكاة بعض من ميزات وإمكانيات البيئة الهدف، بالتالي بعض البرامج الضيفة تحتاج تعديل لتعمل على البيئة الافتراضية. في هذا النمط كل آلة افتراضية لديها فضاء عنوان مستقل بالتالي تطبق الافتراضية على فضاء العنوان. تقوم الآلة الافتراضية بمحاكاة مثائل (Instances) متعددة من بيئة العتاد الصلب المتاحة وأهمها فضاء العنوان. هنا نظام التشغيل كما هو لن يعمل على الآلة الافتراضية وسيحتاج لتعديل، وهنا الفرق عن الافتراضية الكاملة للعتاد الصلب.

هذا النمط كان علامة مميزة فبسبب قصوره في بعض النواحي تم الوصول للافتراضية الكاملة، ويمكن استخدامه لوصف أي نظام تشغيل يوفر فضاء عنوان مستقل للمستخدمين أو للعمليات أو الآلات الافتراضية. وهو نمط أسهل التحقيق، ويوفر آلات افتراضية مفيدة، قوية وقادرة على دعم التطبيقات، وقد نجح فعليا في تحقيق مشاركة موارد الحاسب بين عدة مستخدمين. مقارنة بالافتراضية الكاملة فإن الخلل في الحالات التي تطلب توافقية، من الصعب تقدير أي من الموارد أو الميزات يحتاجها هذا التطبيق أو ذاك. إذا لم تتم محاكاة ميزات محددة، يخفق ذلك التطبيق الذي يستخدم هذه الميزات.

4-3- اللقطات (snapshots)

اللقطة هي حالة الآلة الافتراضية، تمكّن اللقطة من تخزين حالة الآلة الافتراضية في لحظة ثم إعادة استرجاعها لاحقاً، بما يمكن ذلك من تلافي التغييرات التي تحصل. هذه المقدرة تعتبر تقنية مفيدة لعمليات الاسترجاع، عل سبيل المثال، في حالات تنفيذ المهام الحرجة.

تستخدم الآلة الافتراضية أقراص افتراضية لعلميات التخزين، بمثال بسيط، سواقة قرص صلب سعتها 10 جيجا بايت يتم محاكاتها بملف حجمه 10 جيجا بايت. أي استعلام من قبل الآلة الافتراضية لعمليات على قرصها الصلب يترجم كعملية على هذا الملف. في كل مرة يتم أخذ لقطة، يتم إنشاء هكذا ملف. لتتم بعدها عمليات الآلة الافتراضية عليه، والبيانات الجديدة يتم إضافتها لهذا الملف، لنحصل على بنية تشبه بنية المكدّس، المكدّس يحتوي على اللقطة وفيها البيانات من الأقدم للأحدث وهذا المكدس هو منطقياً قرص واحد شامل لكل بيانات حالة الآلة الافتراضية، بهذا الشكل تنشأ لقطة تمثل عمليات حفظ البيانات بشكل تراكمي.

يمكن إضافة مكونات أخرى من الآلة الافتراضية إلى اللقطة، مثل مكونات ذاكرة الوصول العشوائي (RAM)، إعدادات (Bios) وغيرها.

عملية استرجاع اللقطة يجب أن يأخذ بعين الاعتبار إلغاء كافة التعديلات التي أجريت بعد أخذ اللقطة. ثم إنشاء لقطة جديدة تتضمن التعديلات الجديدة تلك.

نظام تشغيل الضيف على الآلة الافتراضية يمكنه إنشاء لقطة تتضمن نسخة عن كل النظام وحفظها في ملف احتياطي، يحفظ كنسخة لعملية استرجاع مستقبلية محتملة.

4-4-التهجير (Migration)

اللقطة التي تم أخذها عن حالة آلة افتراضية يمكن نقلها لحاسب فيزيائي، عندما تتوقف الآلة الافتراضية بشكل مؤقت، تنقل اللقطة، وتشغل على حاسب مضيف جديد وهذا ما يطلق عليه بالتهجير. يمكن أن تتم هذه العملية بسرعة وتمكن من تقديم الآلة الافتراضية لخدمة مستمرة في حال توقف الجهاز الفيزيائي المضيف عن العمل لسبب ما، كالصيانة مثلاً.

4-5- تجاوز الفشل (Failover)

بشكل مشابه لعملية التهجير، تجاوز الفشل يتيح للآلة الافتراضية إمكانية مواصلة عملها في حال توقف الحاسب الفيزيائي المضيف. في هذه الحالة تستمر الآلة الافتراضية بعملها من آخر حالة جيدة كانت عليها، بالاعتماد على ملف البيانات التي تم حفظها عن حالتها عند حدوث الفشل.

4-6-الافتراضية (virtualization) والحوسبة السحابية (Cloud Computing)

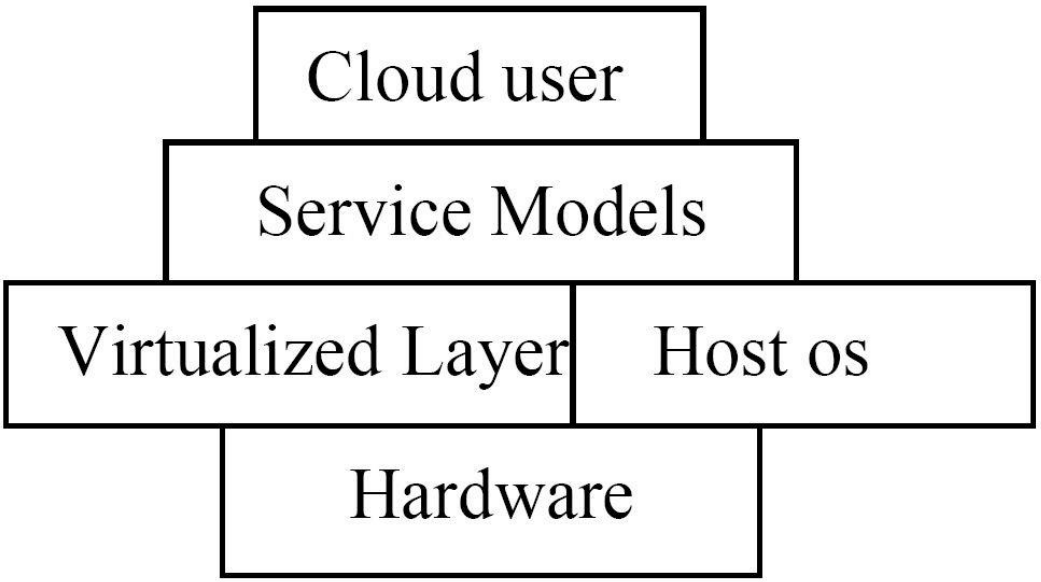
تختلف الافتراضية عن الحوسبة السحابية لان الافتراضية هي برمجية تفصل البنية التحتية الفيزيائية لخلق موارد افتراضية متنوعة. وهي التقنية الأساسية التي تقود الحوسبة السحابية، والحوسبة السحابية تشير للخدمة الناتجة عن الافتراضية، حيث يتم تسليم موارد الحوسبة كخدمة عند الطلب عبر الشبكة[34].

تلعب تقنية الافتراضية دورا رئيسياً في الحوسبة السحابية، لنتخيل مثلا الحوسبة السحابية بدون استخدام تقنية الافتراضية، يمكن للمستخدمين مثلاً تبادل البيانات الموجودة في السحابة ومشاركتها مثل تطبيق برمجي معين أو صور أو أغاني، ولكن مع استخدام تقنية الافتراضية في الحوسبة السحابية أصبح بإمكان المستخدمين مشاركة الموارد الفيزيائية للسحابة من معالجة وذواكر وسعات تخزين وبمعنى آخر استخدام تقنية الافتراضية في الحوسبة السحابية مكّنت السحابة من تجميع كافة أنواع الموارد والبنى التحتية و تسخيرها بشكل كفوء لتقدّم الخدمة التي يطلبها المستخدم بشكل مفيد وبما يتناسب مع عمله.

يحدث لغط في التمييز بين تقنية الافتراضية والحوسبة السحابية لأنهما تعملان معاً لتوفير مختلف أنواع الخدمات. يمكن تبسيط الموضوع بأن الحوسبة السحابية هي مظلة تضم تحتها باقي مفاهيم التقنيات المختلفة ومن ضمنها الافتراضية ومهمتها بالتالي تقديم الخدمات المتنوعة.

4-7-البنية الأساسية لتقنية الافتراضية في الحوسبة السحابية

يتكون نموذج البنية الأساسية لتقنية الافتراضية من: مستخدمين سحابة، ونماذج الخدمة، طبقة الافتراضية ونظام التشغيل المضيف والعتاد الصلب كما هو موضح في الشكل (1-4). ونماذج الخدمة هي نماذج خدمة السحابة المعرفة مسبقاً بأنواعها الثلاثة: البرمجيات كخدمة (SaaS)، منصات العمل كخدمة (PaaS) والبنية التحتية كخدمة (IaaS).



الشكل (4-1) البنية الأساسية لتقنية الافتراضية في الحوسبة السحابية

الفصل الخامس

الآلية المقترحة لتحسين

موازنة الحمل

الفصل الخامس
الآلية المقترحة لتحسين موازنة الحمل

5-1-مقدمة

يوجد العديد من منصات العمل لبناء السحب منها المأجورة ومنها المفتوحة المصدر وتعتبر المنصة (Eucalyptus Platform) واحدة من أفضل عشر منصات العمل المفتوحة المصدر وذلك لسهولة إدارتها وتوفيرها لقابلية التوسع وتم استخدام (Eucalyptus) كمنصة لبناء السحابة التي تمت عليها هذه الدراسة.

5-2-المنصة Eucalyptus

المنصة (Eucalyptus) هي تطبيق برمجي مفتوح المصدر يُستخدم لبناء السحب الخاصة والهجينة وتأتي التسمية اختصاراً للتسمية أداة هندسة الحوسبة لربط برامجك الخاصة بأنظمة مفيدة (Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems) تم تطوير (Eucalyptus) من قبل مجموعة باحثين في جامعة (California Santa Barbara) ويستفيد (Eucalyptus) من بنية الحوسبة والتخزين المتاحة للعمل على بناء البنية التحتية كخدمة (IaaS).

يتميز (Eucalyptus) عن المنصات الأخرى المتوفرة لبناء السحب بسهولة تنصيبه واستخدامه وأيضاً من ناحية الأمان خلال اتصال مكوناته مع بعضها ومن ناحية توافقه مع أشهر المنصات العالمية للحوسبة السحابية وهو ((Amazon Web services (AWS))، فيتيح إمكانية إنشاء سحابة هجينة مع سحابة يوفرها أمازون وذلك من خلال الأدوات المتوافقة مع سحابة أمازون. كما ويمكن استخدامه بدون إجراء تعديلات على البنى التحتية التي يعمل عليها. تملك مكونات (Eucalyptus) واجهات معروفة للتفاعل فيما بينها، وتدعم الاتصالات الآمنة ويتميز بقابلية أن يكون نظام موزع حيث يمكن نشر مكوناته في مكان وجود الموارد، مثلا مكون Walrus المسؤول عن التخزين يمكن وضعه في نفس العنقود مع وسائط التخزين، بينما كل متحكم عنقود يوضع في نفس الشبكة التي يديرها.

يشكل (Eucalyptus) منصة عمل تعتمد على نظام التشغيل (Linux) ويحقق السحابة الخاصة والهجينة بطريقة فعالة ومطورة على البنية التحتية المتاحة ويتم الوصول إلى هذه السحابة الخاصة من قبل المستخدمين عبر شبكات الاتصال الخاصة المتاحة، لذلك تبقى المعطيات الحساسة آمنة من التهديدات الخارجية بفضل وجود أجهزة ومن أهم مكوناته [35]:

1–متحكم السحابة (Cloud Controller [CLC]): هو برنامج مكتوب بلغة (java) يحتوي على قاعدة بيانات تسجل موارد السحابة ومكوناتها. يوجد فقط متحكم سحابة (CLC) واحد في كل سحابة. يتيح هذا المتحكم (CLC) إمكانية إدارة السحابة من خلال واجهة ويب بالإضافة لذلك يقوم المتحكم بمعالجة الطلبات الواردة إلى السحابة وجدولة الموارد وإدارة المستخدمين. بشكل مختصر متحكم السحابة هو المسؤول عن المصادقة، إدارة الحسابات، إعداد التقارير وإدارة الموارد.

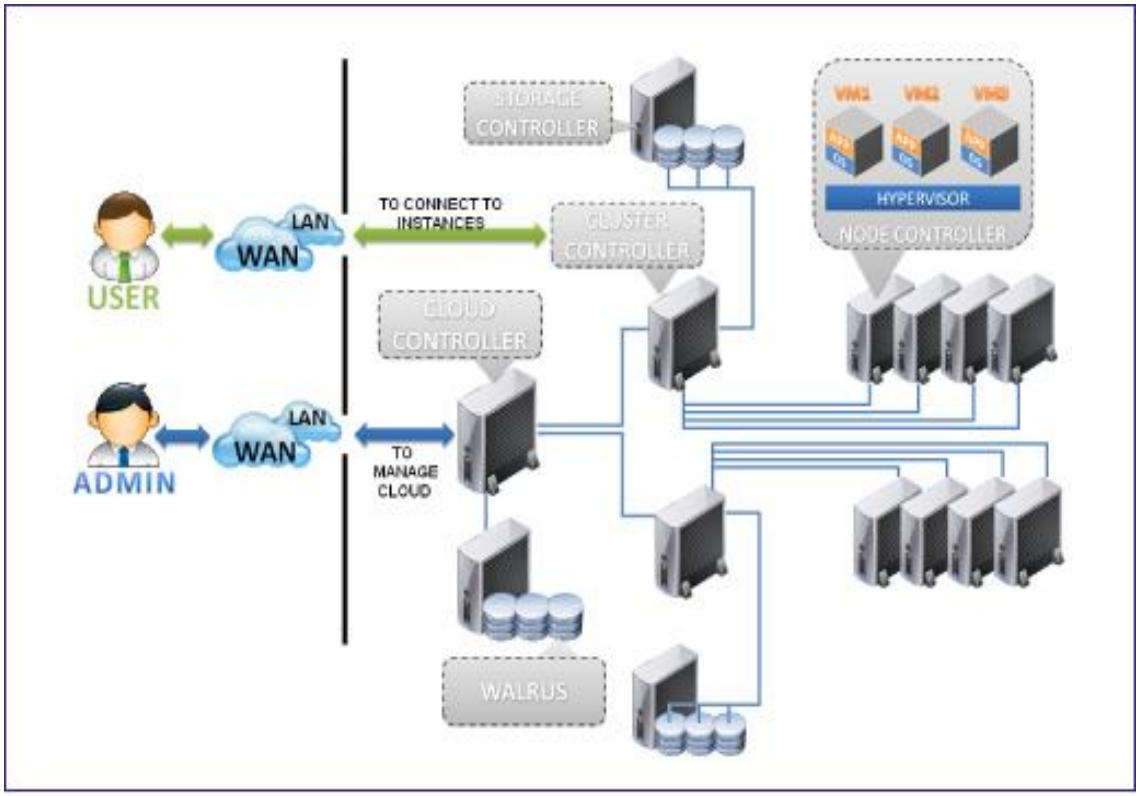
2–متحكم (Walrus): تطبيق مكتوب بلغة (Java) ويوفر خدمة تخزين البيانات، يطبق هذا المتحكم واجهات (REST) و (SOAP) المعيارية، ولا يوجد قيود أو شروط على نوع البيانات التي يتيح هذا المتحكم تخزينها. ومن هذه البيانات (صور الآلات الافتراضية، بيانات التطبيقات).

3–متحكم العنقود (Cluster Controller [CC]): تطبيق مكتوب بلغة (C) ويمثل واجهة العنقود في سحابة (Eucalyptus) والتي تؤمن الاتصال بين متحكمات العقد (NCs) ومتحكم السحابة (CLC). يقوم هذا المتحكم (CC) بجمع البيانات عن متحكمات العقد ويعد التقارير ويقدمها إلى المتحكم (CLC). ويتلقى الطلبات من (CLC) ويجدول نشر الآلات الافتراضية على متحكمات عقد محددة.

4–متحكم التخزين (Storage Controller [SC]): تطبيق مكتوب بلغة (Java) ويتصل بكل من متحكم العقد ومتحكم العنقود ويقوم بإدارة أحجام تخزين البيانات.

5–متحكم العقدة (Node Controller [NC]): تطبيق مكتوب بلغة (C) ويجب أن يتواجد على كل عقدة في السحابة. هو مسؤول عن استضافة مثيلات (Instances) الآلات الافتراضية والإدارة الشبكية لها. ويحمّل صور الآلات الافتراضية من (Walrus) من أجل تشغيل المثيلات.

متحكم العقدة (NC) يستعلم من العقدة عن الموارد الفيزيائية (عدد نوى المعالج، حجم الذاكرة، والمساحة الحرة على قرص التخزين) ويجمع المعلومات عن حالة المثيلات العاملة على العقدة. نظرياً طالما لا حد لعدد متحكمات العقد في العنقود، لا يوجد بالتالي حد للأداء الممكن توفيره. ويوضح الشكل (1-5) بنية سحابة (Eucalyptus).



الشكل (5-1) بنية سحابة Eucalyptus

5-3- خوارزميات جدولة نشر الآلات الافتراضية في سحابة (Eucalyptus)

لكسب الحدّ الأعلى من استخدام الحوسبة السحابية، يجب أن يصمّم المطوّرون الآليات التي تحسّن الاستخدام الأمثل للموارد الفيزيائية. شكلت الآلات الافتراضية (VMs) قضية مهمة لأنه من خلال تقنية الافتراضية (Virtualization) تكون البنى التحتية للسحابة الحاسوبية قابلة للتوسع. نقدم في هذه الأطروحة خوارزميات الجدولة المستخدمة في المنصة (Eucalyptus): الخوارزمية الشرهة (Greedy algorithm) وخوارزمية البيان الدوّار (Round Robin algorithm).

5-3-1-الخوارزمية الشرهة (Greedy Algorithm)

استعملت الخوارزمية الشرهة بشكل افتراضي لجدولة الآلات الافتراضية في (Eucalyptus). وكانت السياسة الوحيدة للجدولة وبقيت قيد الاستعمال لوقت طويل. وبعد أن بدأت السحابة بالتطوّر دخلت سياسات جدولة أكثر تعقيداً حيّز التنفيذ. إنّ الخوارزمية الشرهة بسيطة جداً، حيث تستعمل الخوارزمية الشرهة العقدة الأولى لتشغيل الآلات الافتراضية بالموارد المناسبة المخصصة وتستمر العقدة الأولى المعروفة باستقبال الآلات الافتراضية حتى تنتهي مواردها فيتم البدء بتوجيه الآلات الافتراضية إلى العقدة التالية. هذا يعني بأنّ الخوارزمية الشرهة ترهق العقدة قبل أن تنتقل إلى العقدة التالية. الفائدة

الرئيسية للخوارزمية الشرهة هي البساطة حيث أن تخصيص الآلات الافتراضية لا يتطلّب أيّ معالجة معقّدة والعائق الرئيسي يتجلّى بالاستخدام المنخفض للموارد المتاحة وتم التغلب على عائق الخوارزمية الشرهة بتطبيق خوارزمية البيان الدوار.

5-3-2-خوارزمية البيان الدوار (Round Robin Algorithm)

في هذه الخوارزمية، يخصّص المجدول (Scheduler) آلة افتراضية واحدة إلى العقدة بطريقة دورية. يبدأ المجدول (Scheduler) مع عقدة وينتقل إلى العقدة التالية، بعد تخصيص الالة الافتراضية إلى تلك العقدة. يكرّر حتى يخصص كلّ العقد على الأقل بآلة الافتراضية واحدة وبعد ذلك يعود المجدول (Scheduler) إلى العقدة الأولى مرة ثانية. في هذه الخوارزمية لا ينتظر المجدول استهلاك مصادر العقدة قبل الانتقال إلى العقدة القادمة. الفائدة الرئيسية لهذه الخوارزمية بأنّها تستعمل كلّ الموارد في تنفيذ الطلب بالإضافة إلى تخصيص عدد متساوي من الآلات الافتراضية إلى كلّ العقد وبالتالي تضمن الإنصاف. ولكن هذه الطريقة لا تعير اهتمام لموارد كل عقدة.

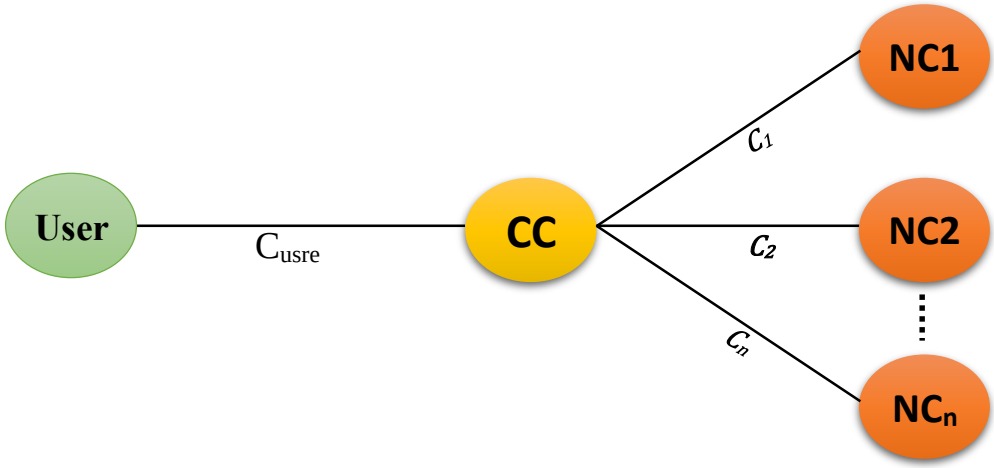
5-4-آلية موازنة حمل نشر الآلات الافتراضية المقترحة

موازنة الحمل هي عبارة عن آلية لتوزيع التدفّق (traffic) على عدّة مخدمات افتراضية خاصّة عبرَ تقسيم آلية المعالجة إلى عدّة أجهزة مما يضمن استقراراً أفضل وتعاملاً أكثر سلاسة مع الأخطاء [36].

الآلية المقترحة تطبّق لتحديد متحكم العقدة الأنسب لتشغيل الآلة الافتراضية (VM). يتكون النظام المدروس من نظام سحابة (Eucalyptus) إضافةً لمراقب حالة الشبكة (NSM) المقترح. سيتم في بنية النظام المدروس وضع متحكمات سحابة (Eucalyptus) على مخدم واحد وسيتم إضافة المتحكم (NSM) على نفس المخدم.

يُظهر الشكل (5-2) البنية التنظيمية لسحابة (Eucalyptus) وهي بنية هرمية بسيطة مع متحكم عنقود (cluster controller) في قمة البنية. متحكم العقدة (NC) يرمز لجهاز فيزيائي يشغّل الآلات الافتراضية. في التطبيق العملي، يمكن نشر متحكمات العقدة (NCs) في شبكة واحدة أو عدة شبكات فرعية. من خلال ذلك الاعتبار للبنية، كل اتصال شبكي بين (NC) و (CC) سيختلف من ناحية استخدام عرض النطاق (Bandwidth). لكل مستخدم يتصل مع المتحكم (CC) سعة اتصال معينة يرمز لها (C_{user}). لنفترض وجود (n) عقدة (NC_n) تعطى بالمجموعة $N=\{N_1,N_2,..\ N_n\}$، حيث كل

سعة اتصال بين متحكم العقدة و متحكم العنقود يرمز لها بالرمز(C_n) ومجموعة سعات اتصال لكل متحكمات العقد (NCs) مع متحكم العنقود يعطى من خلال ($C=\{C_1, C_2,\ldots C_n\}$).



الشكل (2-5) البنية التنظيمية لسحابة Eucalyptus

سيتم في هذا القسم شرح الآلية المقترحة لاختيار أفضل متحكم عقدة (NC) لتشغيل الآلات الافتراضية (VMs) بما يحقق ضمان جودة خدمة الشبكة (QoS).

تعتمد الآلية المقترحة على غرض رئيسي بتجنب حالات عنق الزجاجة خلال تدفق البيانات. الميزة الرئيسية لهذه الآلية هي حساب كيفية تحقيق الموازنة بين (C_{user}) و(C_n). القيمة الأولية التي توضع هي المعدل لأعظم سعة اتصال بين المستخدم ومتحكم العنقود (CC)، يرمز لها (C_{user}). عندئذ، يتم حساب المعدل الوسطي لكافة سعات الاتصال (C_n) بناء على قياسات متحكم مراقبة حالة الشبكة (NSM) المضاف، ويعطى المعدل الوسطي من خلال العلاقة ($C_{Avg} = \frac{\sum_{i=1}^{n} C_i}{n}$). بناء على هاتين القيمتين، فإن القواعد في اختيار (NC) هي كالتالي:

1- إذا $C_{user} > C_{Avg}$، سيتم اختيار متحكم العقدة (NC) صاحب أعلى سعة اتصال [max[Cn].

2- إذا $C_{user} < C_{Avg}$ ، سيتم اختيار متحكم العقدة (NC) صاحب سعة الاتصال ([max([Cn] < C_{Avg}).

3- إذا $C_{user} \approx C_{Avg}$ ، سيتم اختيار متحكم العقدة (NC) صاحب سعة الاتصال ([min([Cn] >= C_{Avg}).

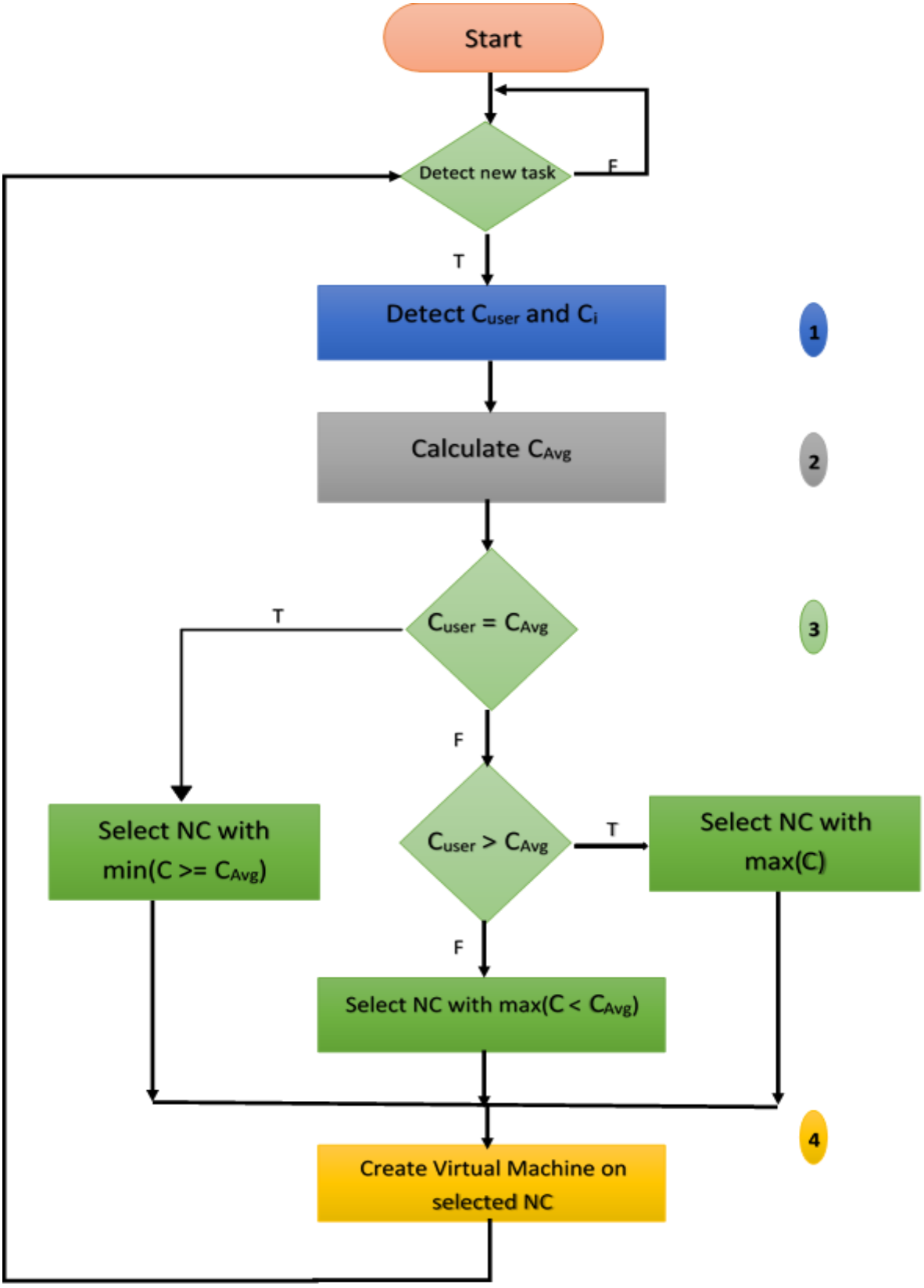
-خوارزمية اختيار (NC) هي كالتالي:

1- الخطوة الأولى: عندما يتلقى متحكم مراقبة حالة الشبكة (NSM) المضاف طلب من متحكم العنقود (CC)، يبدأ بمراقبة وتسجيل حالة استخدام الشبكة بين متحكم العنقود (CC) وكل متحكم عقدة (NC) وتُعطى بالتسمية (Cn)، وبين المستخدمين ومتحكم العنقود (CC) وتُعطى بالتسمية (C_{user}).

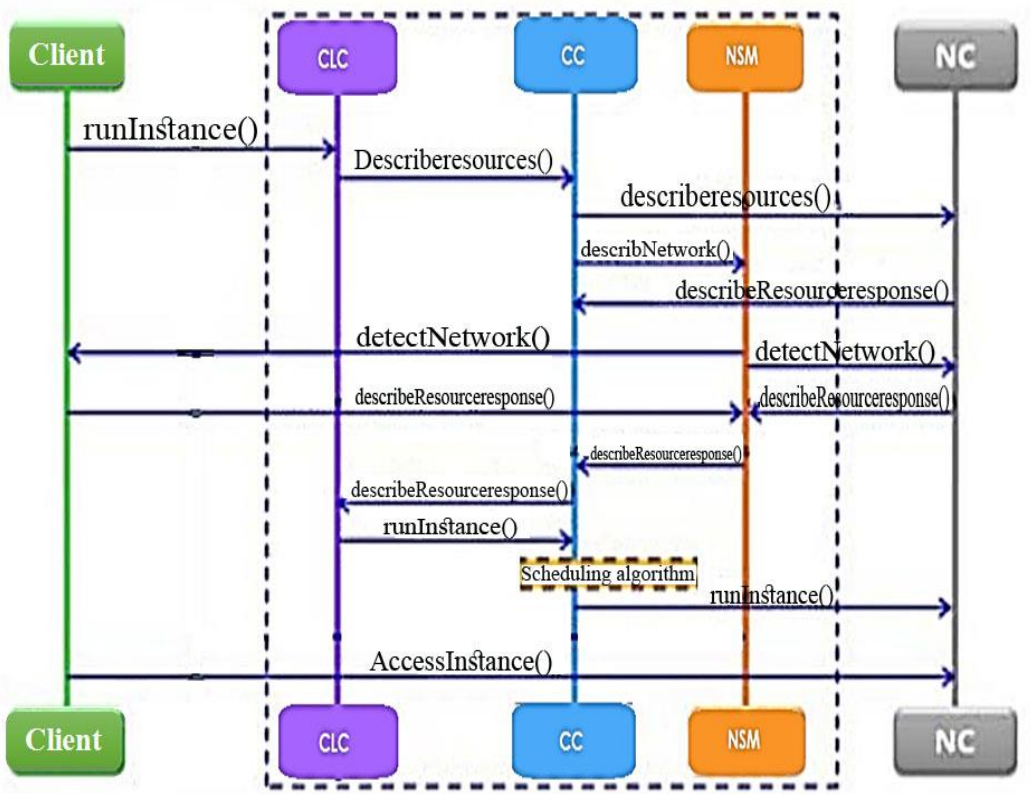
2- الخطوة الثانية: بعد أن يحصل متحكم مراقبة حالة الشبكة (NSM) المضاف على جميع النتائج، يحسب السعة الوسطية (C_{Avg}) لجميع سعات اتصال متحكمات العقد المتوفرة Cn.

3- الخطوة الثالثة: يقوم متحكم مراقبة حالة الشبكة (NSM) عندها بالمقارنة بين (C_{user}، C_{Avg}) ويختار متحكم العقدة (NC) الأنسب بناءً على الآلية السابقة.

4- الخطوة الرابعة: يقوم متحكم العنقود (CC) بتشغيل الآلة الافتراضية على متحكم العقدة الذي تم اختياره. يظهر الشكل (3-5) مخطط تدفقي لاختيار NC المناسبة لتشغيل VM عليها، ويبين الملحق رقم (B) الشيفرة البرمجية لمراقب حالة الشبكة NSM.



الشكل (3-5) المخطط التدفقي لاختيار NC مناسبة لتشغيل الآلة الافتراضية عليها



الشكل (4-5) توضيح مبسط لإجرائية تشغيل الالة الافتراضية

يُظهر الشكل رقم (4-5) توضيح مبسط لإجرائية تشغيل الآلة الافتراضية (VM)، أولاً عندما يريد المستخدم الوصول لخدمات سحابة (Eucalyptus) يتم تشغيل مثيل (instance) أيّ طلب تعليمة (runInstance()) ضمن متحكم السحابة (CLC)، يختار متحكم السحابة (CLC) بدوره أول متحكم عنقود (CC) يملك كمية كافية من الموارد وذلك بالأمر (Describeresources()) ليعلم كمية الموارد الموجودة. المتحكم (CC) يستعلم من كل (NC) من خلال التعليمة (describeresources())، يستجيب متحكم العقدة (NC) لمتحكم العنقود (CC) ويعيد المعلومات عن الموارد المتوافرة من خلال التعليمة (describeResourceresponse()). بشكل متزامن، يرسل متحكم العنقود (CC) طلب (describeNetwork()) للمتحكم المضاف (NSM) للاستعلام عن سعات الاتصال. بعد أن يتلقى المتحكم الطلب، يرسل طلب (detectNetwork()) لجمع المعلومات عن الشبكة بين (CC) مع (NCs) وبين (CC) مع المستخدم. عندئذ متحكم العنقود (CC) ينفذ عمليات لاختيار متحكم العقدة (NC) المناسب لنشر الآلة الافتراضية (VM) وفقاً للآلية المقترحة.

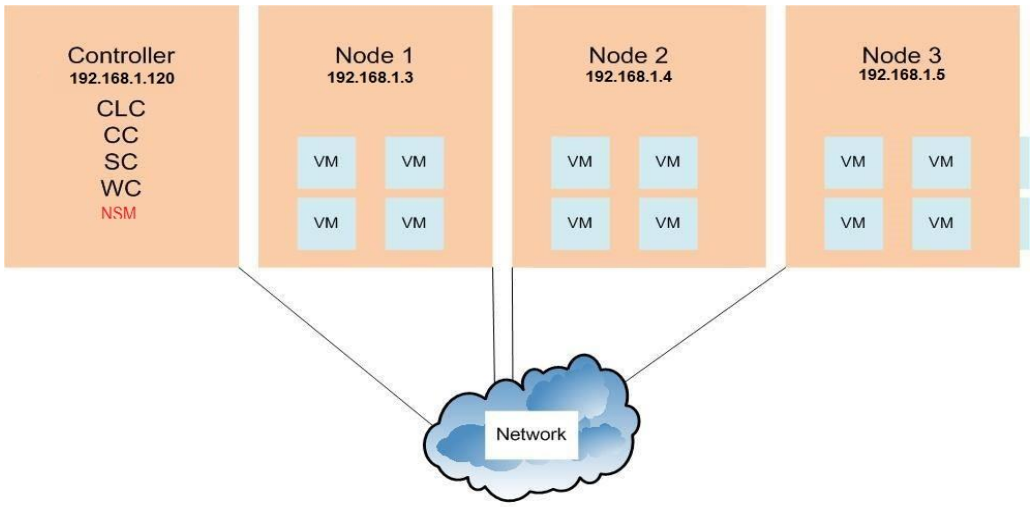
5-5-تصميم النظام وتحليل الأداء

5-5-1-تطبيق النظام

تم إنجاز هذه الدراسة باستخدام مخدّم يحتوي على (CLC, CC, SC, Walrus, NSM) وثلاثة أجهزة تمثل (NC1,NC2,NC3) وذلك لتحقيق بيئة مناسبة لسحابة (Eucalyptus). يُظهر الجدول 5-1- أجهزة نظام Eucalyptus المطبق والشكل (5-5) إعدادات النظام على شبكة فرعية واحدة:

الجدول 5-1-أجهزة نظام Eucalyptus المطبق

	Controller	Node1	Node2	Node3
Functionality	**CLC, CC, and Walrus, *NSM***	**NC**	**NC**	**NC**
Host Name	**Cluster**	**EucNC1**	**EucNC2**	**EucNC3**
CC Address	**192.168.1.120**			
Gateway	**192.168.1.1**	**192.168.1.2**	**192.168.1.2**	**192.168.1.2**
Instances public address	**192.168.1.200-192.168.1.220**			
Cluster Name	**Euccluster**			
Hardware device	**PC(Intel Core2 Dou CPU E7500 @ 2.93Ghz, 4 GB RAM, Intel 1350 T2(NIC)**			
Software Device	**OS: Ubuntu 10.04, cloud Software: Eucalyptus 1.6.2**			



الشكل (5-5) إعدادات النظام على شبكة فرعية

5-5-2-محاكاة النظام

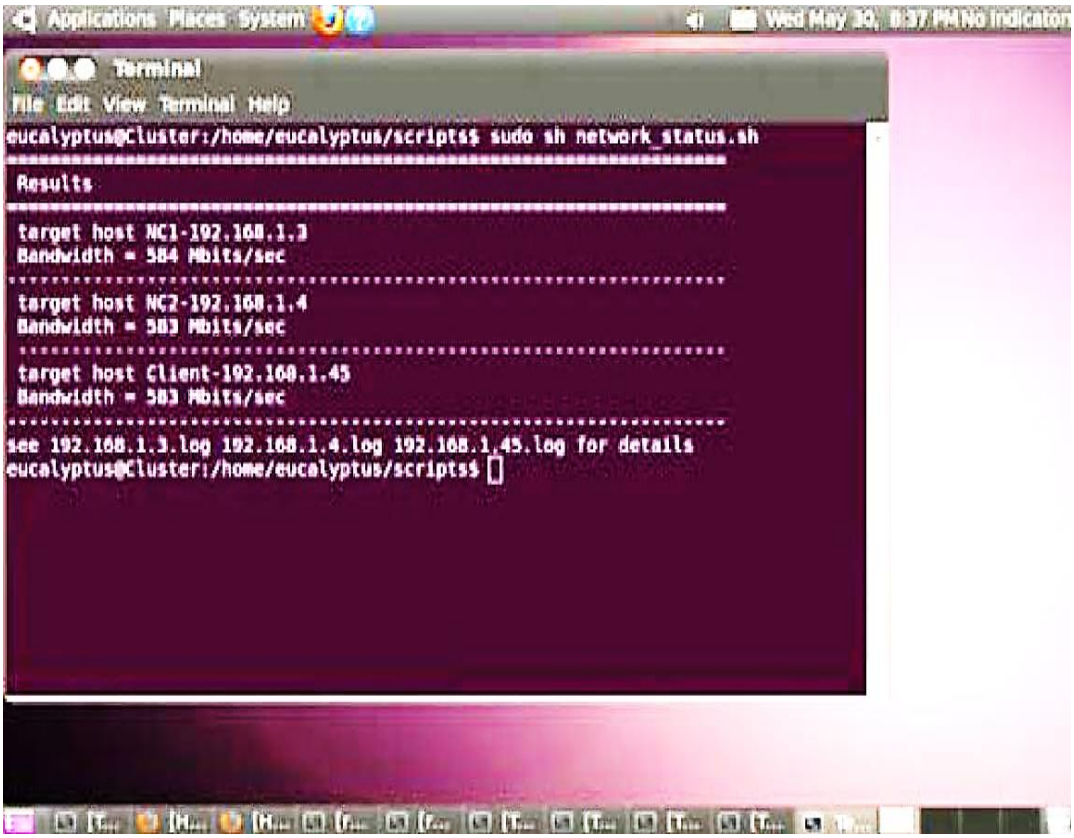
يظهر الشكل رقم (5-6) واجهة إدارة (Eucalyptus)، ضمن صفحة الصلاحيات، حيث يستطيع المدير والمستخدمين تحميل ملف الصلاحيات (credentials) لاستخدام الأداة (Euca2ools) لإدارة السحابة وطلب الخدمات.



الشكل (5-6) واجهة إدارة Eucalyptus

تعرض صفحة (images) كل ملفات (image) الخاصة بأنظمة التشغيل المحمّلة والمتوفرة على السحابة وتكون من نوع (ISO) والتي سيتم القراءة منها عند تشغيل أي آلة افتراضية، تظهر صفحة المستخدم معلومات المستخدم المدير. الصفحة الأكثر أهمية هي صفحة الإعدادات والتي يستخدمها المدير لوضع إعدادات (CLC). تظهر صفحة الخدمات معلومات متعلقة بنظام (Eucalyptus) وصفحة (extra) تسرد بعض الأدوات المتوافقة مع (Eucalyptus).

للحصول على سعات الاتصال (Cu-cc , CN(i)) تم بناء متحكم مراقبة حالة الشبكة بجانب متحكم السحابة ومتحكم العنقود، يظهر الشكل رقم (5-7) عمل المتحكم المضاف، حيث يقوم بقياس جميع الاتصالات من متحكم العنقود (CC) ليحصل على شروط الاتصال لكل من متحكمات العقد (NCs) المتوافرة من جهة والمستخدمين من جهة أخرى.



الشكل (5-7) عمليات متحكم مراقبة حالة الشبكة المضاف

5-5-3-تحليل الأداء

تم تحليل أداء النظام المدروس باستخدام أدوات قياس زمني متوفرة في نظام (Ubuntu) في حالتين دراسيتين:

الحالة الدراسية الأولى: متحكم العقدة في نفس الشبكة الفرعية.

الحالة الدراسية الثانية: متحكم العقدة في شبكة فرعية مختلفة.

كل حالة تعمل مع ثلاث آليات لنشر الآلات الافتراضية:

1- (Round Robin Mechanism [ROROM]).

2- (Greedy Mechanism [GrM]).

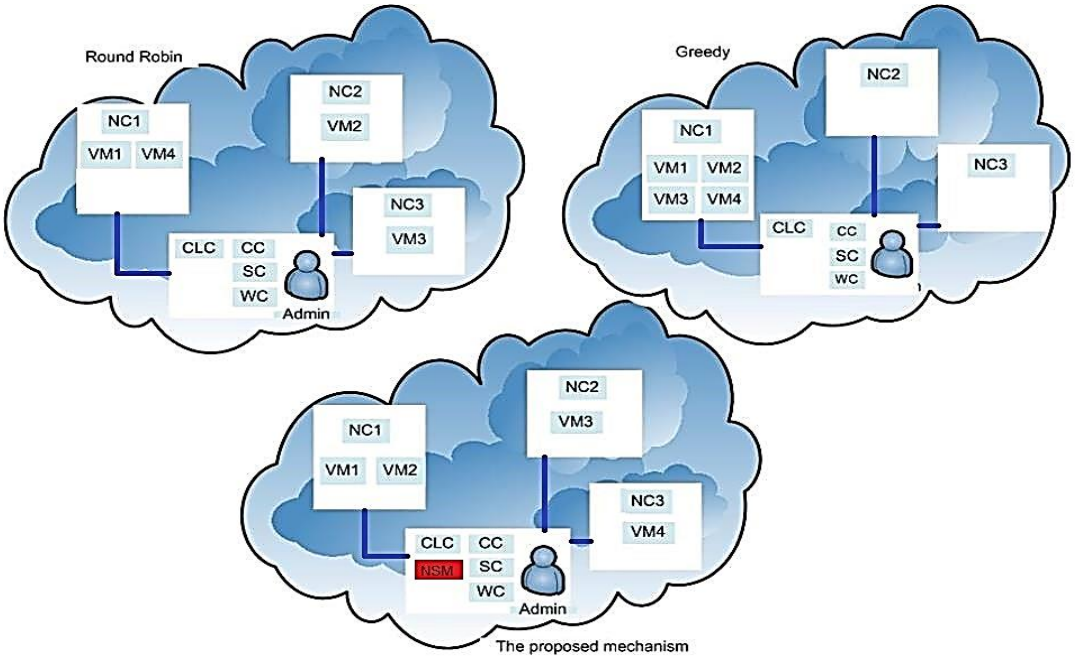
3- آلية موازنة حمل نشر الآلات الافتراضية المقترحة.

في هذه الدراسة تم إنشاء أربع آلات افتراضية (VMs) وتم مراقبة مكان توضع كل (VM) في كل حالة مدروسة وتم تنفيذ جميع الآلات الافتراضية باستخدام ملف (ISO) واحد وذلك لتحديد كيفية تأثير توضع (VM) على أداء النظام. تم تحميل ملف (ISO) إلى الآلة الافتراضية كل 10 ثواني، في البداية يحمّل الملف إلى الآلة الافتراضية الأولى ثم بعد 10 ثواني يحمّل إلى الآلة الافتراضية الثانية وبعد 10 ثواني إلى الآلة الافتراضية الثالثة وعند الثانية 30 إلى الآلة الافتراضية الرابعة.

5-5-4-النتائج التحليلية هي كالتالي

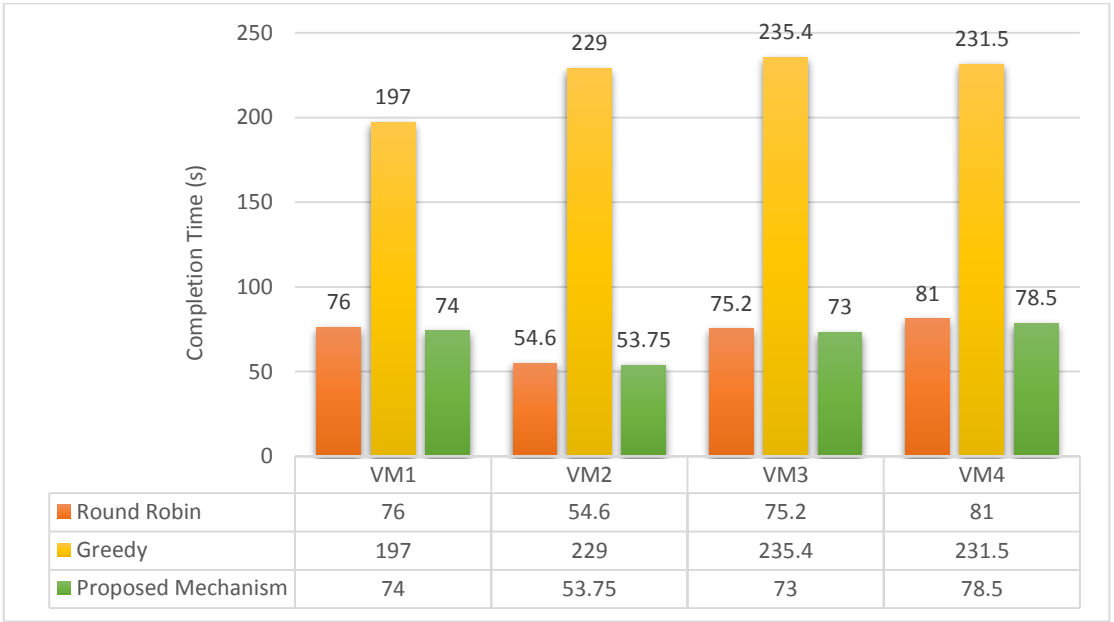
- **الحالة الدراسية الأولى:** متحكم العقدة في نفس الشبكة الفرعية:

يُظهر الشكل رقم (5-8) توضع الآلات الافتراضية مع آليات الجدولة المختلفة. الآلية الأولى (RoRoM) تتوزع الآلات الافتراضية في كل متحكمات العقد (NCs) حيث (VM1) و (VM4) تتوضع في متحكم العقدة الأولى (NC1) و (VM2) وَ (VM3) تتوضع في متحكم العقدة الثانية (NC2) ومتحكم العقدة الثالثة (NC3) على التوالي. بينما مع الآلية الثانية (GrM) تتوضع الآلات الافتراضية الأربعة في متحكم عقدة واحد. أما الآلية الأخيرة وهي الآلية المقترحة في هذه الدراسة فإن توضع الآلات الافتراضية في متحكمات العقد هو نفسه في حالة (RoRom) ولكن بتسلسل مختلف كما يوضح الشكل رقم (5-8).



الشكل (5-8) توضع الآلات الافتراضية مع آليات الجدولة المختلفة للحالة الدراسية الأولى

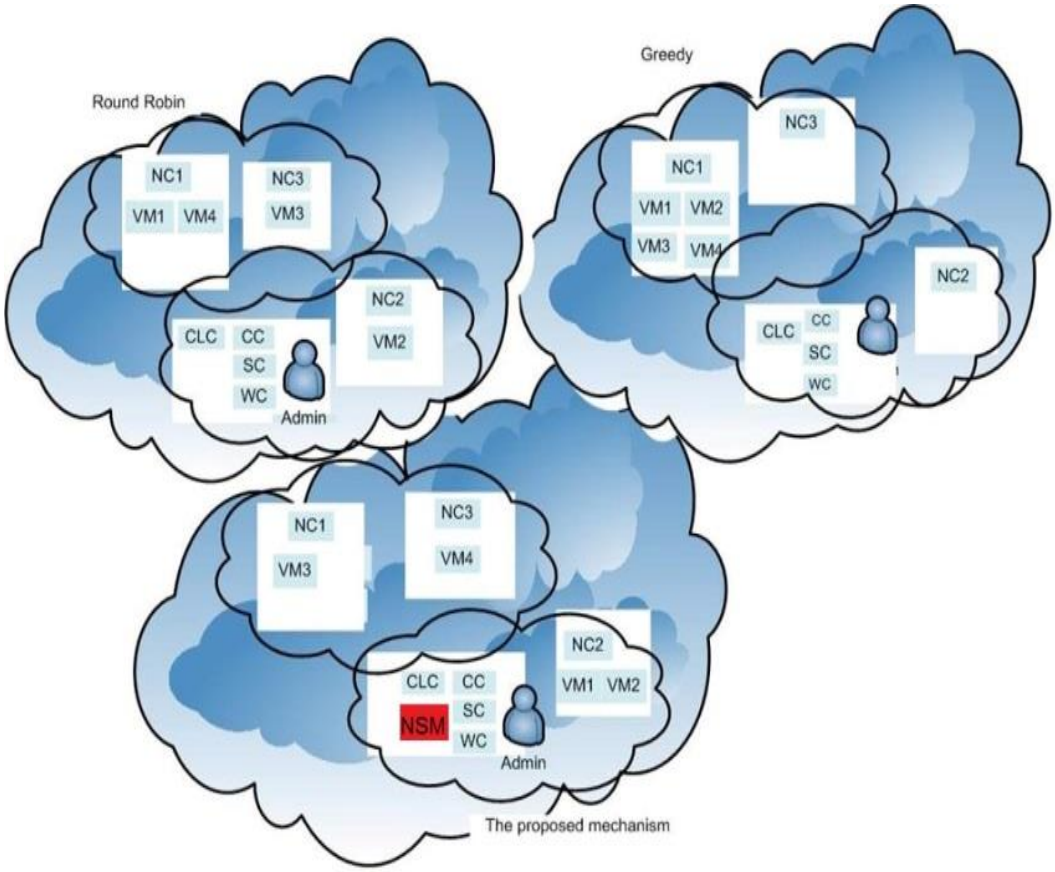
يُظهر الشكل رقم (5-9) وقت الاكتمال لتحميل ملف (ISO) في الحالة الأولى حيث أن وقت الاكتمال هو نفسه تقريباً في حالة آلية (RoRom) والآلية المقترحة لكن الآلية المقترحة أسرع بثواني قليلة، والفرق هو 1.89 ثانية في المتوسط. بينما مقارنةً بالزمن في حالة الآلية (GrM) فإن وقت الاكتمال الوسطي للآلية المقترحة أقل بـ 153.41 ثانية.



الشكل (5-9) مخطط وقت اكتمال تشغيل كل VM في الحالة الدراسية الأولى

- **الحالة الدراسية الثانية:** متحكم العقدة في شبكة فرعية مختلفة

في هذه الحالة، متحكم العنقود (CC) و متحكم العقدة (NC2) يتوضعان في نفس الشبكة الفرعية ومتحكمي العقدتين (NC3،NC1) يتوضعان في شبكة فرعية أخرى. يُظهر الشكل (10-5) توضع الآلات الافتراضية مع آليات الجدولة المختلفة. أولاً، مع آلية (RoRom) تتوضع الآلات الافتراضية في كل متحكمات العقد (NCs) حيث (VM1) و (VM4) تتوضع في متحكم العقدة الأولى (NC1) والذي هو في شبكة فرعية مختلفة عن شبكة متحكم العنقود وتتوضع (VM2) في متحكم العقدة الثانية (NC2) والذي هو في نفس الشبكة الفرعية لمتحكم العنقود وتتوضع (VM3) في متحكم العقدة الثالثة (NC3) الذي هو في نفس شبكة (NC1) الفرعية. بينما في آلية (GrM) تتوضع الآلات الافتراضية الأربعة في متحكم عقدة واحد (NC1) والذي هو على شبكة فرعية مختلفة عن شبكة متحكم العنقود. أما مع الآلية المقترحة باستخدام (NSM) تتوضع كل من (VM1 ، VM2) على المتحكم (NC2) والذي هو على نفس الشبكة الفرعية لمتحكم العنقود أما (VM3) و (VM4) تتوضعان على المتحكمين (NC1) و (NC3) على التوالي الموجودين على شبكة فرعية مختلفة.

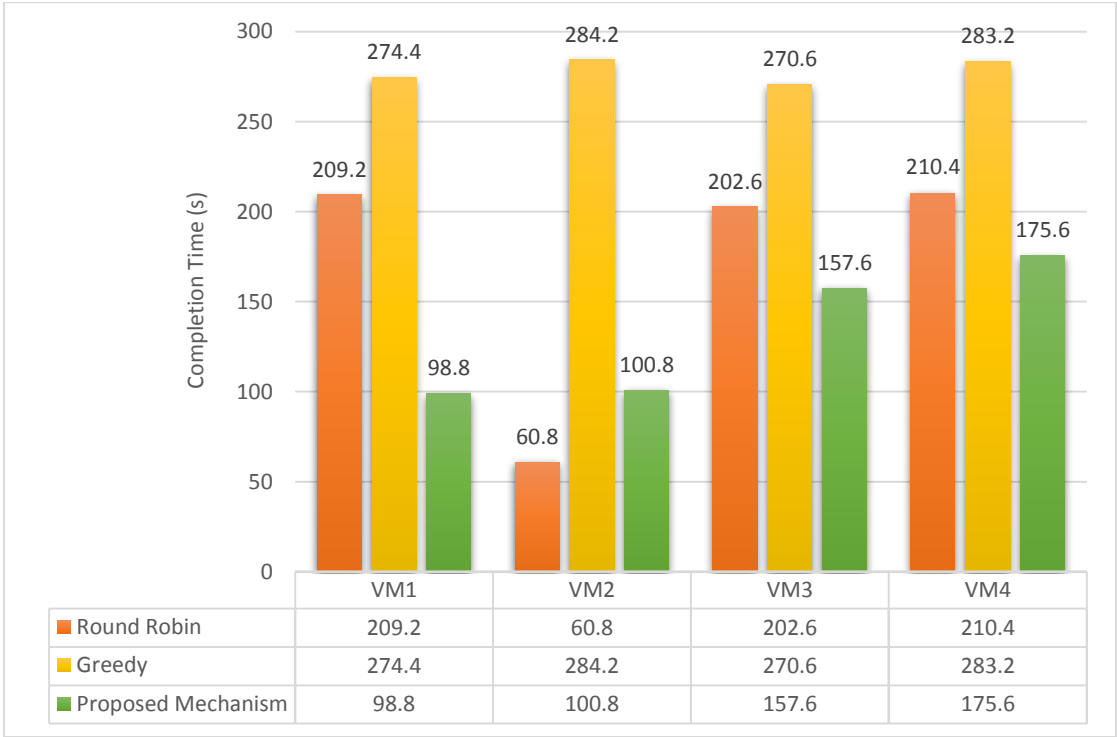


الشكل (10-5) توضع الآلات الافتراضية مع آليات الجدولة المختلفة في الحالة الدراسية الثانية

باستخدام آلية (RoRom) فان (VM1, VM3, VM4) نشرت على شبكة فرعية مختلفة عن شبكة متحكم العنقود بالتالي كان زمن الاستكمال الوسطي هو 207 ثواني، بينما (VM2) التي نشرت على نفس الشبكة الفرعية لمتحكم العنقود استغرق زمن الاستكمال فقط 60.80 ثانية.

أما بالنسبة لآلية (GrM) ومقارنةً مع نتائج الحالة الأولى فان زمن الاستكمال الوسطي يتزايد من 223.23 ثانية إلى 278.10 ثانية وذلك لأن متحكم العقدة الذي تم اختياره لنشر الآلات الافتراضية يتوضع على شبكة فرعية مختلفة عن الشبكة الفرعية لمتحكم العنقود.

وفي الآلية المقترحة باستخدام متحكم مراقبة حالة الشبكة (NSM) تتوضع كل من (VM1, VM2) على متحكم العقدة الثانية (NC2) الموجود على نفس شبكة متحكم العنقود الفرعية فكانت نتائج زمن الاستكمال الوسطي هي 99.80 ثانية أما (VM3, VM4) تتوضع متحكمات عقد موجودة على شبكة فرعية مختلفة عن شبكة متحكم العنقود الفرعية فكانت نتائج زمن الاستكمال الوسطي 166.55 ثانية، كما هو مبين في الشكل رقم (5-11).



الشكل (5-11) مخطط وقت اكتمال تشغيل كل VM في الحالة الدراسية الثانية

5-6-الخاتمة

بسبب التطور المتسارع لتقنية الحوسبة السحابية، فإن عدد المستخدمين لهذه التقنية يزداد بشكل دراماتيكي، وتعتبر مهمة تحديد وتشغيل الآلات الافتراضية (VMs) واحدة من المهام الأساسية لتقنية الحوسبة السحابية. تعتمد الآليات التقليدية لنشر الآلات الافتراضية فقط على الموارد الفيزيائية مثل حجم الذواكر وسرعة المعالجة وسعة التخزين. تقدم هذه الدراسة آلية جدولة جديدة بدراسة سعات نطاق الشبكة وذلك لاختيار متحكم عقدة (NC) مناسب لتشغيل الآلة الافتراضية (VM). النظام المدروس يتضمن البنية التحتية لسحابة Eucalyptus بالإضافة إلى متحكم مراقبة حالة الشبكة (NSM) الذي تمت إضافته ومكونات الشبكة الفيزيائية. المتحكم المضاف (NSM) يراقب استخدام الشبكة ثم يرسل المعلومات إلى متحكم العنقود (CC) الذي يقوم بدوره باختيار متحكم العقدة (NC) المناسب لتشغيل الآلة الافتراضية (VM).

الفصل السادس

النتائج والمقترحات والعمل المستقبلي

الفصل السادس
النتائج والمقترحات والعمل في المستقبل

6-1-النتائج

أصبحت الحوسبة السحابية مؤخراً واحدة من أكثر القضايا أهمية في مجال تقنية المعلومات .وتقوم الحوسبة السحابية على فكرة رئيسية وهي الاستفادة من الموارد المشتركة والتي تقاس تكلفتها بقدر استخدامها عبر الإنترنت. هذه التقنية الجديدة جاءت بوعود كبيرة بإمكانية حل مشكلة التكلفة المرتفعة للموارد إضافةً إلى حل مشاكل العناء والجهد المبذول في إدارة هذه الموارد.

الحوسبة السحابية مفهوم واسع وموازنة الحمل تلعب دوراً هاماً في تحسين جودة وكفاءة استخدام الحوسبة السحابية، وتحسين آلية نشر الآلات الافتراضية في بنية الحوسبة السحابية يؤدي إلى رفع كفاءة استثمار الموارد الفيزيائية للسحابة وبالتالي يؤدي إلى تحسين موازنة الحمل في الحوسبة السحابية، ومن خلال دراسة الحالات السابقة (متحكم العقدة على نفس الشبكة لمتحكم العنقود، ومتحكم العقدة على شبكة فرعية مختلفة عن شبكة متحكم العنقود) تبين لنا مايلي:

1- من خلال تحليل نتائج الحالة الدراسية الأولى (متحكم العقدة على نفس الشبكة لمتحكم العنقود) وعند توزيع الآلات الافتراضية على متحكمات العقد في كل من الخوارزمية الشرهة وخوارزمية البيان الدوار (GrM, RoRoM) بالإضافة إلى الآلية المقترحة لتوضع الآلات الافتراضية نجد أن زمن الاستكمال في الآلية المقترحة هو نفسه تقريباً بالنسبة لحالة خوارزمية البيان الدوار (RoRoM) بتحسن بسيط بضع ثواني، بينما التحسن واضح وكبير بالنسبة لحالة الخوارزمية الشرهة (GrM) وهو تقريباً 153 ثانية.

2- عند تحليل نتائج الحالة الدراسية الثانية (متحكم العقدة في شبكة فرعية مختلفة عن شبكة متحكم العنقود) وتوزيع الآلات الافتراضية في متحكمات العقد نجد انه في حالة خوارزمية البيان الدوار (RoRoM) زمن الاستكمال الوسطي هو تقريبا 170 ثانية أما في حالة الآلية المقترحة فهو 133 ثانية، وفي حالة الخوارزمية الشرهة (GrM) فإن زمن الاستكمال الوسطي يرتفع إلى 278 ثانية تقريباً، في النتيجة فإن آلية موازنة الحمل المقترحة أفضل من الآليات المثبتة مسبقاً (GrM, RoRoM) ومقدار التحسن في زمن الاستكمال الوسطي يكون بمعدل 37 ثانية عن خوارزمية (RoRoM) و 145 ثانية عن خوارزمية (GrM) وهذا واضح في المخططات البيانية.

3- إن ســعة نطاق الشــبكة في الآلية المقترحة لاختيار متحكم العقدة هو أفضــل من الآليات المثبتة سابقاً.

4- كذلك الأمر بالنسبة لنشر الآلات الافتراضية في الشبكة الأساسية أو الفرعية فإن الآلية المقترحة أفضل من خلال المرونة في عملية النشر.

5- إن إضافة آلات افتراضية جديدة إلى الشبكات الفرعية أو الأساسية لا يُنقص من سرعة هذه الشبكات.

6-2-آفاق العمل المستقبلي

إن مفهوم الحوسبة السحابية مفهوم بدأ حديثاً وتطبيقاته العلمية والأكاديمية ذات مجال واسع في تطوير نظم المعلومات وخاصة بالنسبة للمصادر المفتوحة، دراسة الافتراضية على مستوى نظام التشغيل بحيث يمكن استضافة بيانات افتراضية عديدة على نفس نظام التشغيل أو استضافة تطبيقات فردية في بيئة معزولة عن نظام التشغيل (التطبيقات المحمولة).

الملحق –A–

إنشاء الحوسبة السحابية الخاصة باستخدام البنية مفتوحة المصدر
Building a Private Cloud Infrastructure Using Open source

إنشاء سحابة Eucalyptus الخاصة باستخدام نظام التشغيل **Ubuntu Server 10.04 Enterprise** في هذا الشرح سوف نرى الخطوات اللازمة لبناء سحابة حاسوبية خاصة، حيث يمكن أن نرى بعد الانتهاء من بنائها كيف ندير (Image)، مجموعات الأمان، مراقبة الموارد ونشر المثيلات ضمن السحابة الخاصة، اخترنا لهذا الشرح نظام التشغيل (Ubuntu) بسبب السهولة وسرعة النشر.

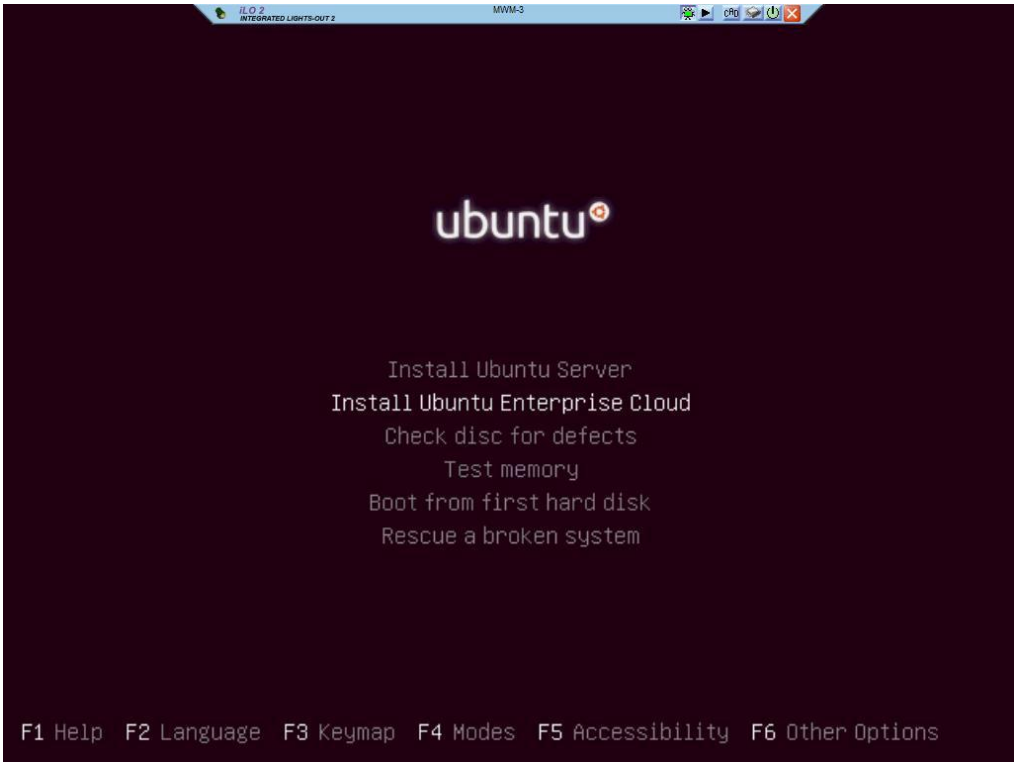
التحضير:

في حالتنا سنستعمل مخدمان، واحد كجهاز متحكم السحابة والآخر كعقدة سحابة. سيتم تشغيل مثيلات (حالات) السحابة على العقدة، وإذا كانت الأنظمة غير متطابقة لذلك سنختار نظام متعدد المعالجات وذاكرة كجهاز متحكم بالعقد. هذا يتيح لنا مجال أكبر للتوسع في السحابة عندما نريد إضافة حالات. سنقوم باستخدام تكوين الشبكة الافتراضية (Managed-No VLAN) الذي يزوّد بعنوان (IP) ديناميكي مخصص للآلات الافتراضية وتسمح لنا بالتحكم بمرور البيانات ببناء (IpTables) المعروف بمجموعات الأمان.

ملاحظة: النمط الآخر من إعدادات الشبكة المعروف بالنمط "المدار Managed" يوفر طريقة ذات ميزة إضافية من شبكة العزلة VM.

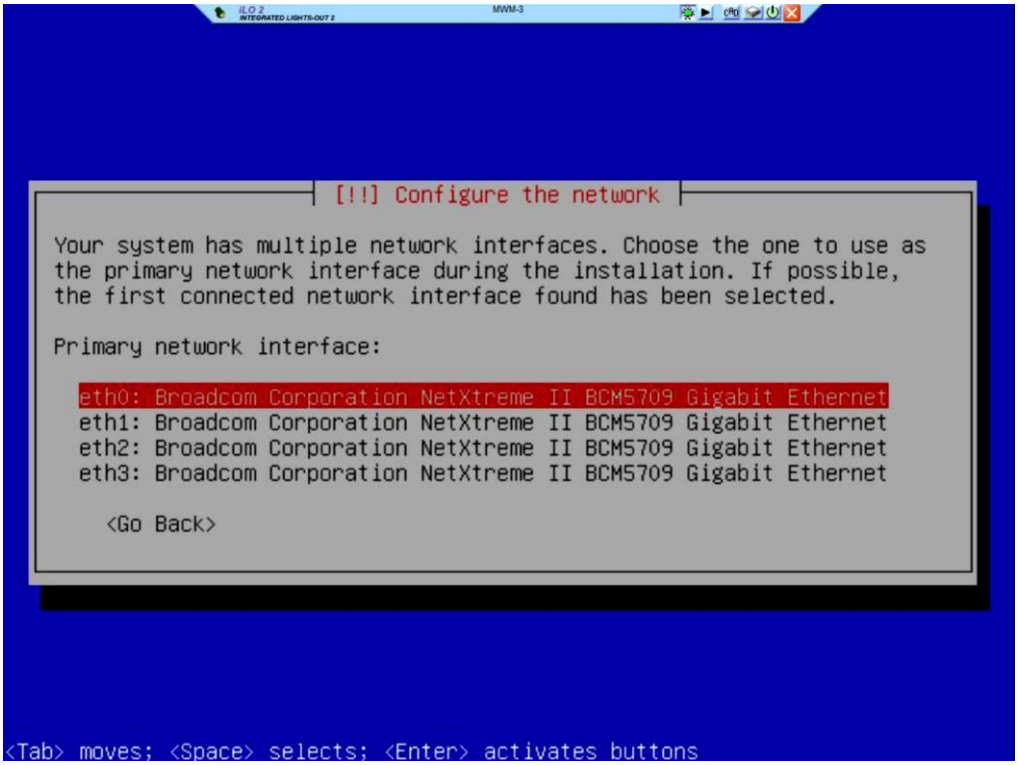
البدء – بناء متحكم السحابة:

أولاً– سوف نبني متحكم السحابة من قرص إقلاع Ubuntu 10.04 Server cdrom واختيار "Install Ubuntu Enterprise Cloud"من القائمة كما هو مبين في الشكل (A-1):



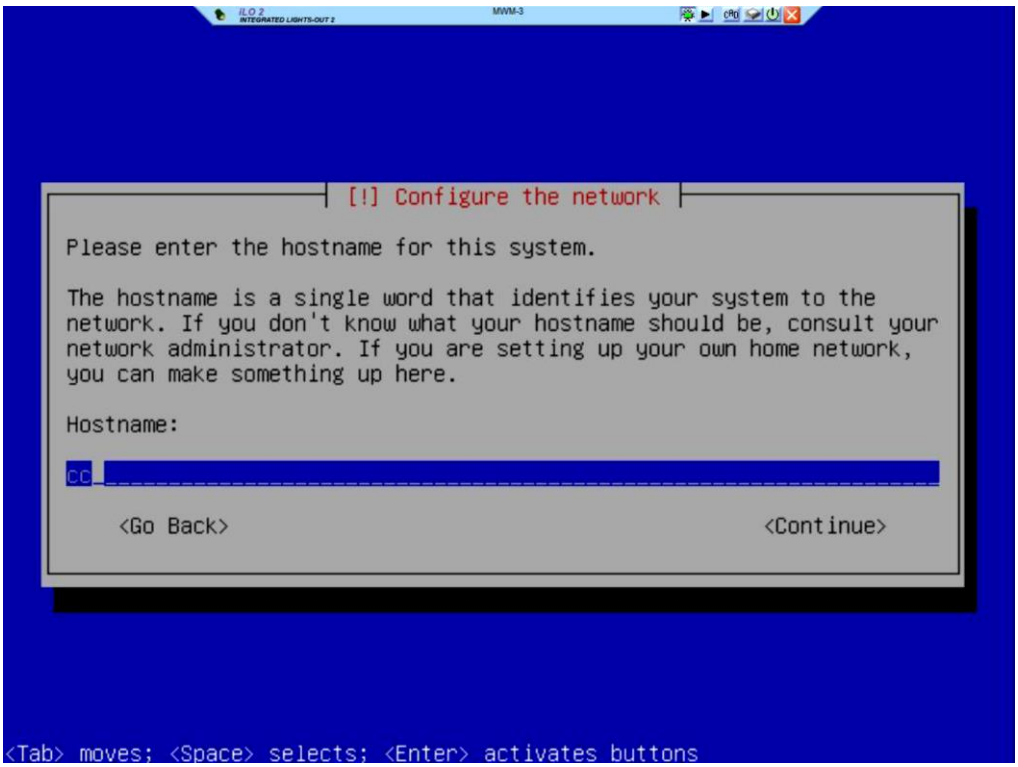
الشكل (A-1) الإقلاع من قرص Ubuntu

بعد اختيار اللغة الملائمة، واختيار لوحة المفاتيح والبلد، سنعمل على إعداد الشبكة. من أجل هذه التجربة سوف نقوم باستخدام eth0 لكلا متحكم السحابة والعقدة كما هو مبين في الشكل (A-2).



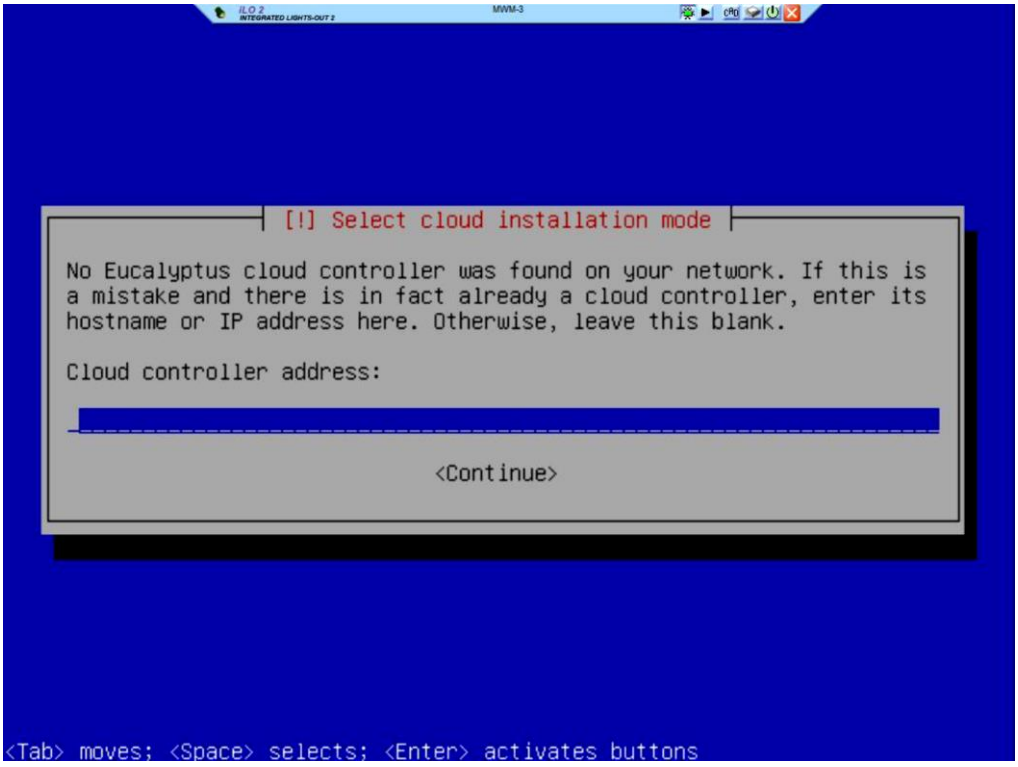
الشكل (A-2) اختيار الشبكة eth0

ويظهر في الشكل (A-3) تعيين اسم المضيف، لقد اخترنا الاسم هنا CC من أجل متحكم السحابة.



الشكل (A-3) اختيار اسم متحكم السحابة

ولأننا لا نملك في هذه الشبكة متحكم سحابة نختار الاستمرار في الشاشة التالية كما في الشكل (A-4):

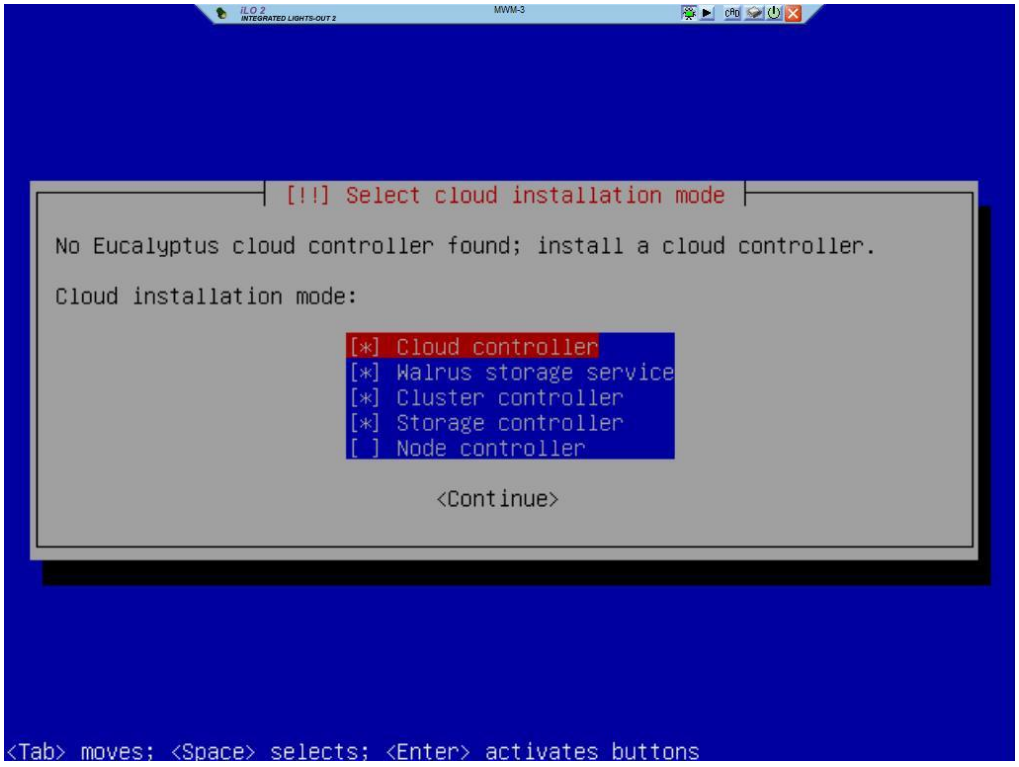


الشكل (A-4) الاستمرار إلى الشاشة التالية

هنا يجب أن نختار دور هذا المخدم الذي سيعمل في سحابتنا. في أكبر و/ أو أكثر التجهيزات تعقيداً، ويمكن تقسيم كلّ هذه الوظائف المدرجة في هذه القائمة على مخدمات فعلية المنفصلة.

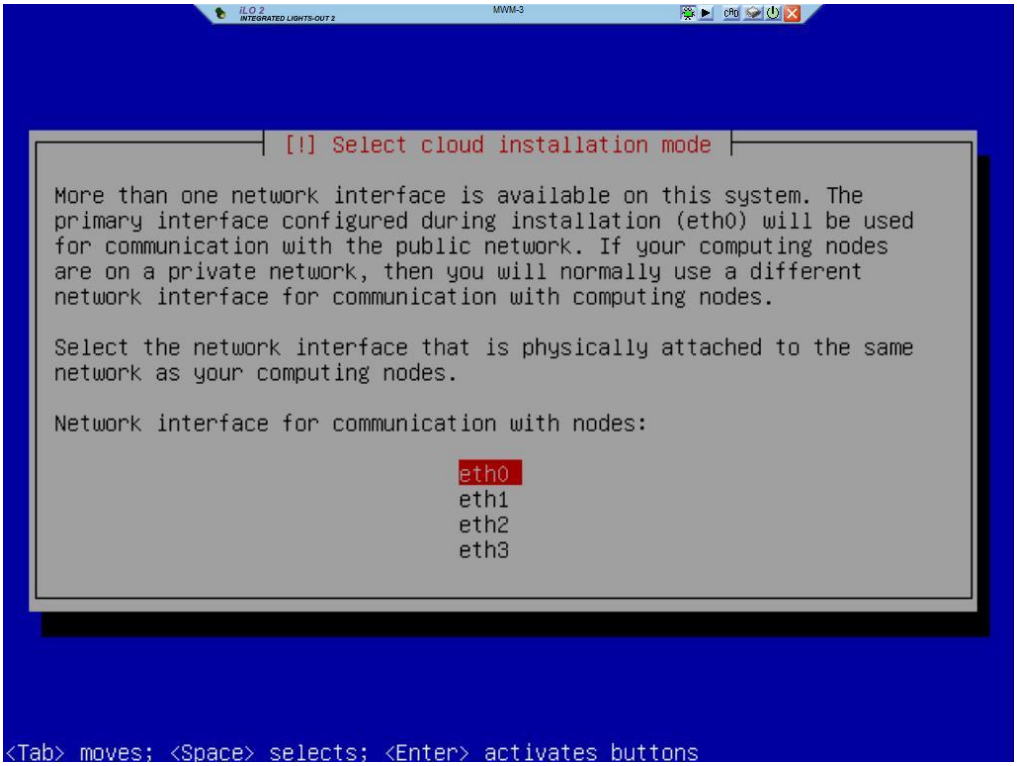
سيكون لسحابة Eucalyptus الخاصّة متحكم سحابة واحد، ولكن قد يكون هناك متحكمات عناقيد متعدّدة ضمن السحابة، ومتحكمات عقدة متعدّدة تجهز إلى كلّ متحكم عنقودي. Walrus متحكم التخزين هو عبارة عن مكوّن تخزين البيانات Eucalyptus، والذي يشابه خدمة تخزين البسيطة لأمازون (S3).

سنستعمل عنقود مفرد ونقبل تركيب نمط السحابة الأساسية لمتحكم السحابة، خدمة تخزين Walrus، متحكم العنود، ومتحكم التخزين كما يبين الشكل (A-5).



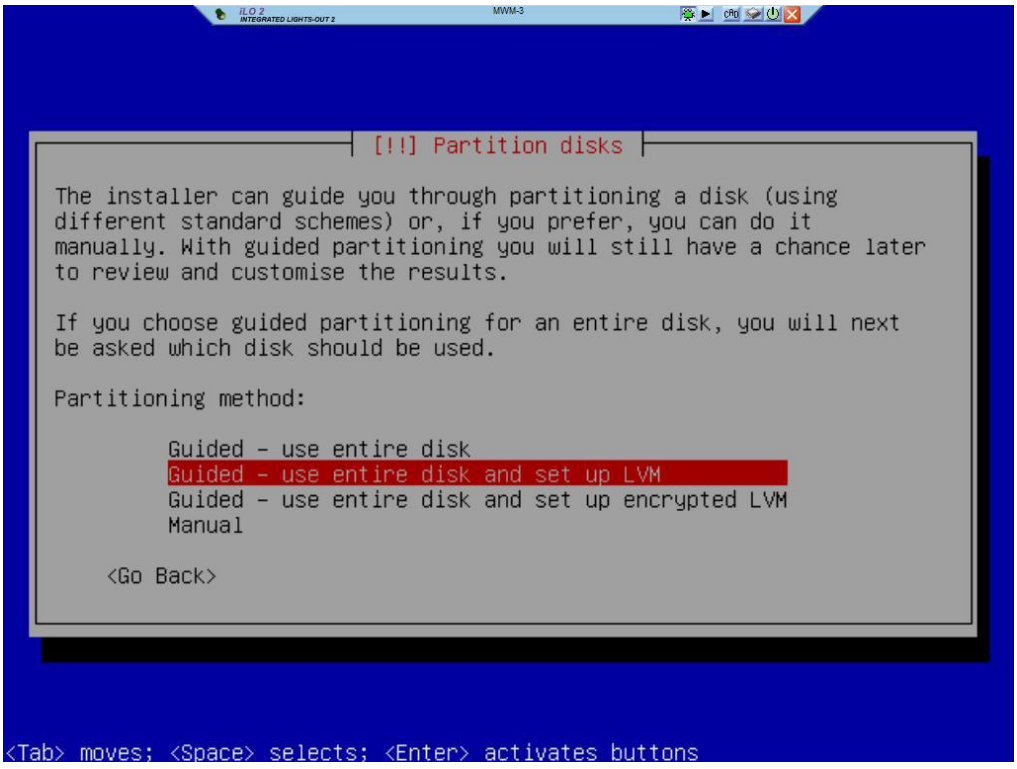
الشكل(A-5) اختيار متحكم السحابة

في هذه التجربة سوف نستخدم واجهة eth0 للاتصال بالشبكة العامة والاتصال مع العقدة كما يبين الشكل (A-6).



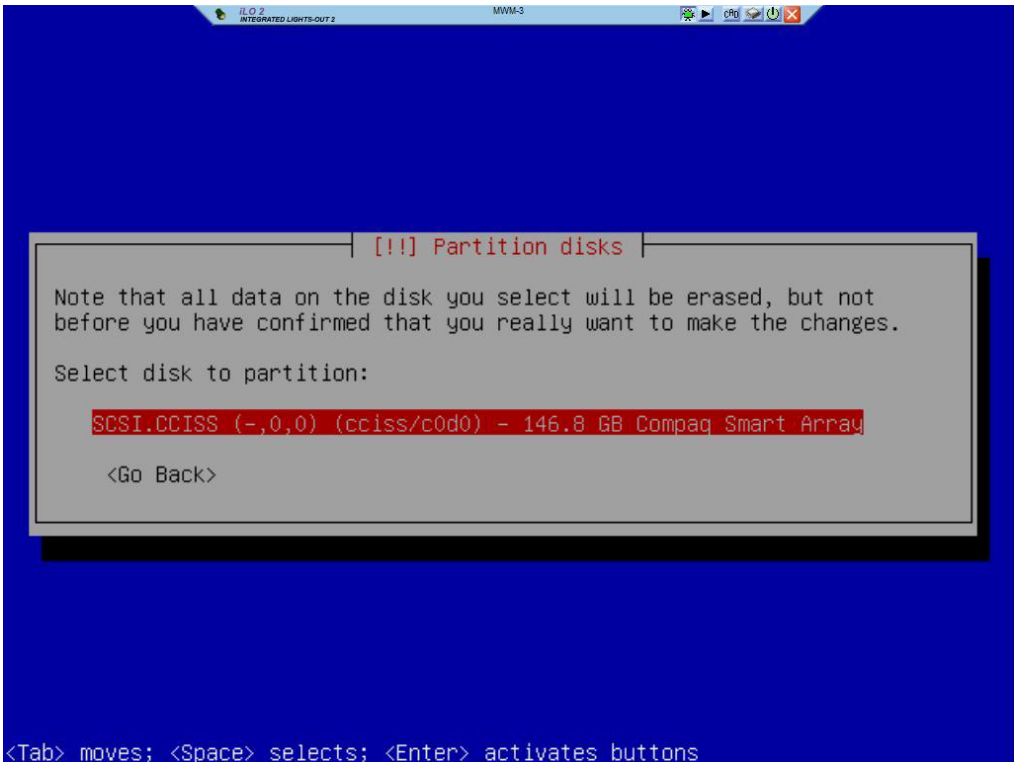
الشكل (A-6) استخدم واجهة eth0 للاتصال بالشبكة العامة

في الشكل (A-7) سوف نوافق في الشاشات القادمة على الخيارات الافتراضية لتقسيم القرص الصلب.



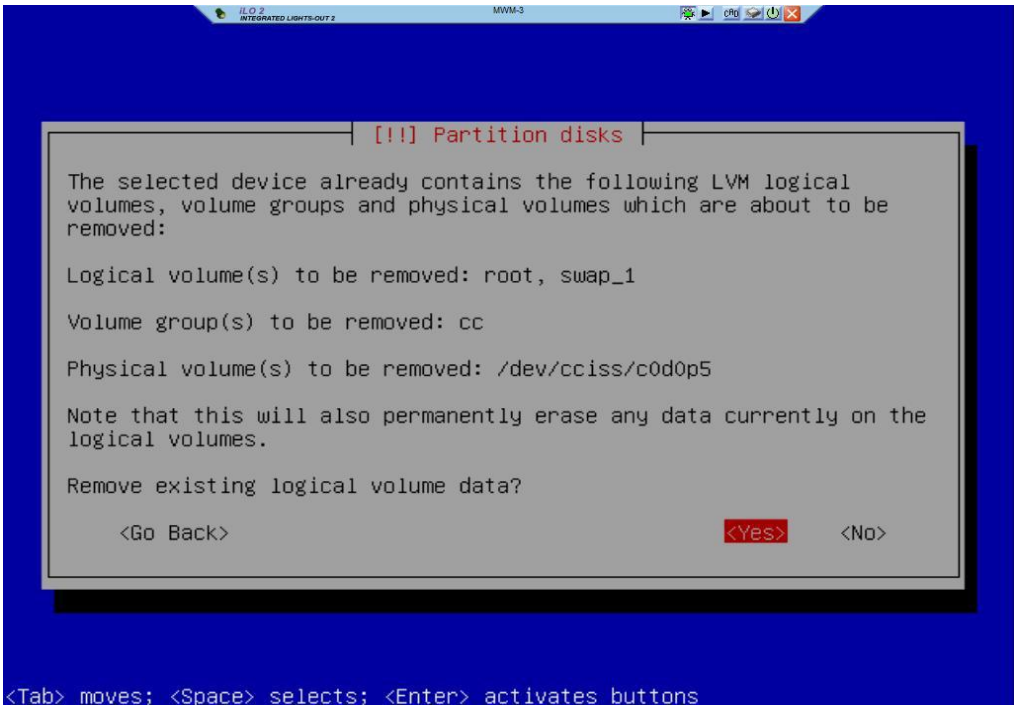
الشكل (A-7) خيارات تقسيم القرص الصلب

لدينا قرص صلب واحد لتنصيب نظام التشغيلUbuntu ، ونختار من البيوس RAID 1+0 كما في الشكل (A-8).



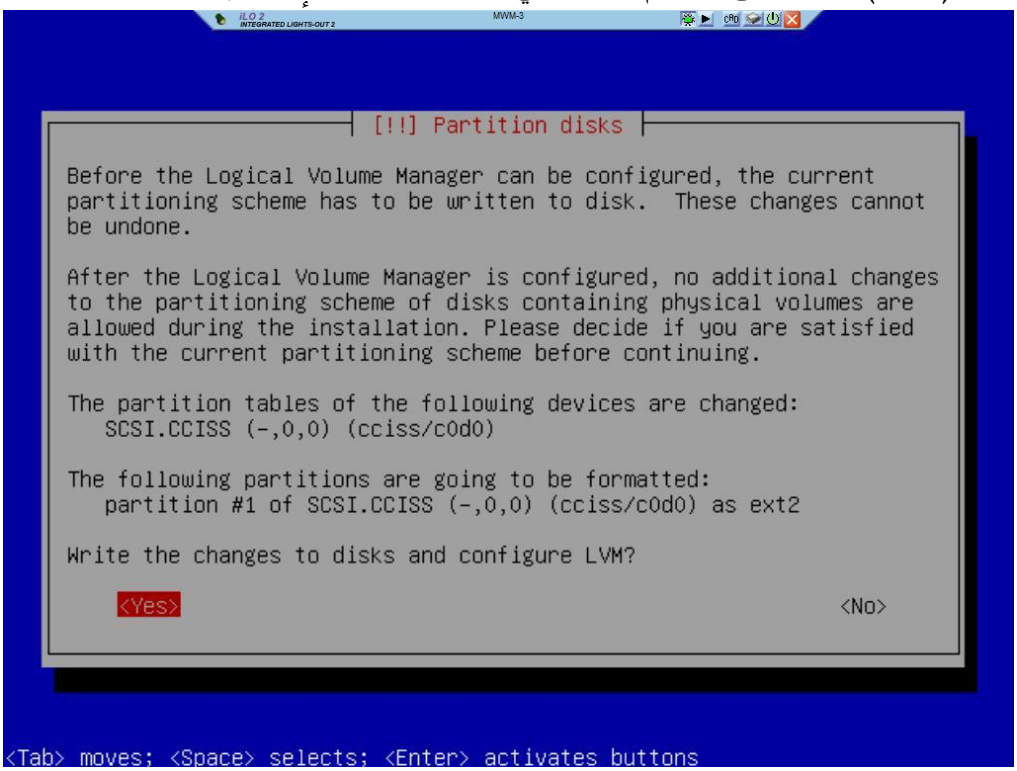
الشكل (A-8) اختيار القرص الصلب

عندها سوف يتم حذف كافة البيانات من على القرص الصلب، كما هو مبين في الشكل (A-9).



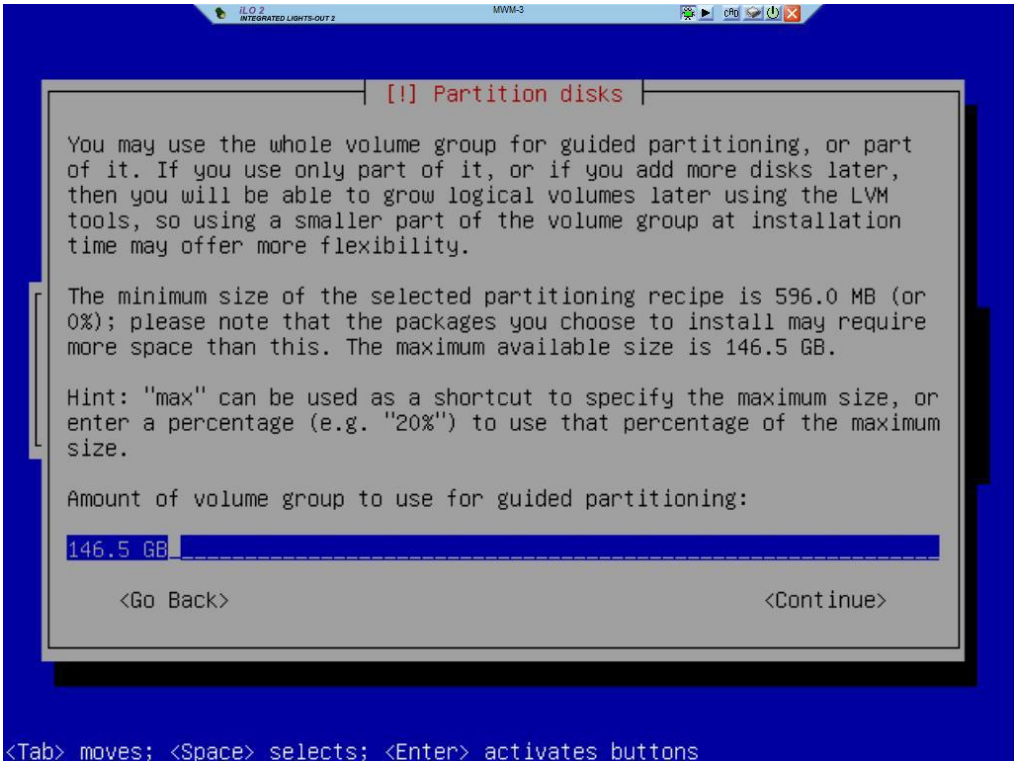
الشكل (A-9) رسالة تتطلب الموافقة على حذف كافة المعلومات من القرص الصلب

في الشكل (A-10) نوافق على التقسيم الافتراضي للقرص الصلب وإعداد إدارة نسخة Linux.



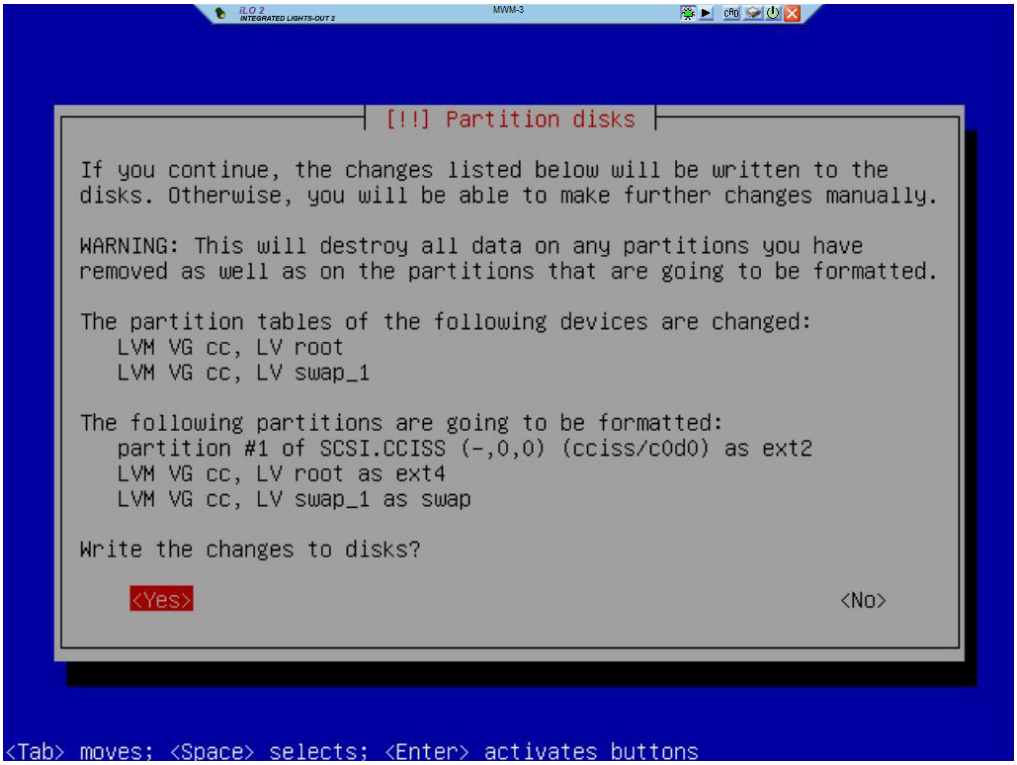
الشكل (A-10) الموافقة على التقسيم الافتراضي للقرص الصلب

في هذه التجربة سوف نستخدم كامل حجم القرص الصلب كما هو مبين في الشاشة الشكل (A-11).



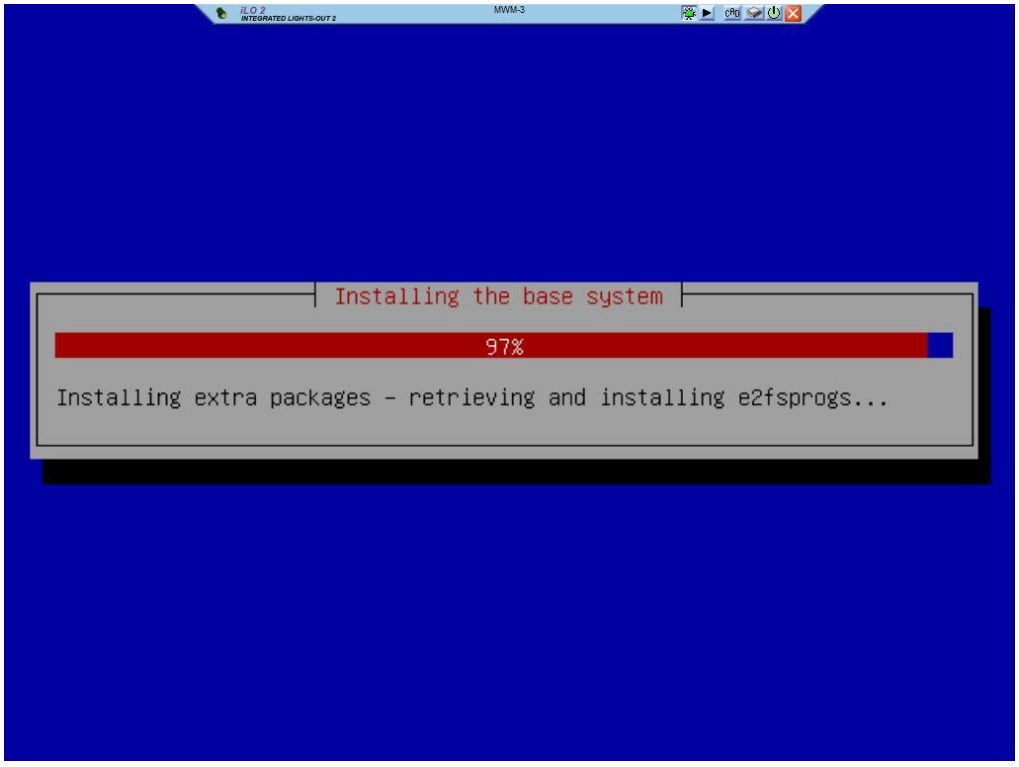
الشكل (A-11) استخدام كامل مساحة القرص الصلب

بعدها سوف نحفظ التغيرات الجديدة على القرص الصلب كما هو مبين في الشاشة الشكل (A-12).



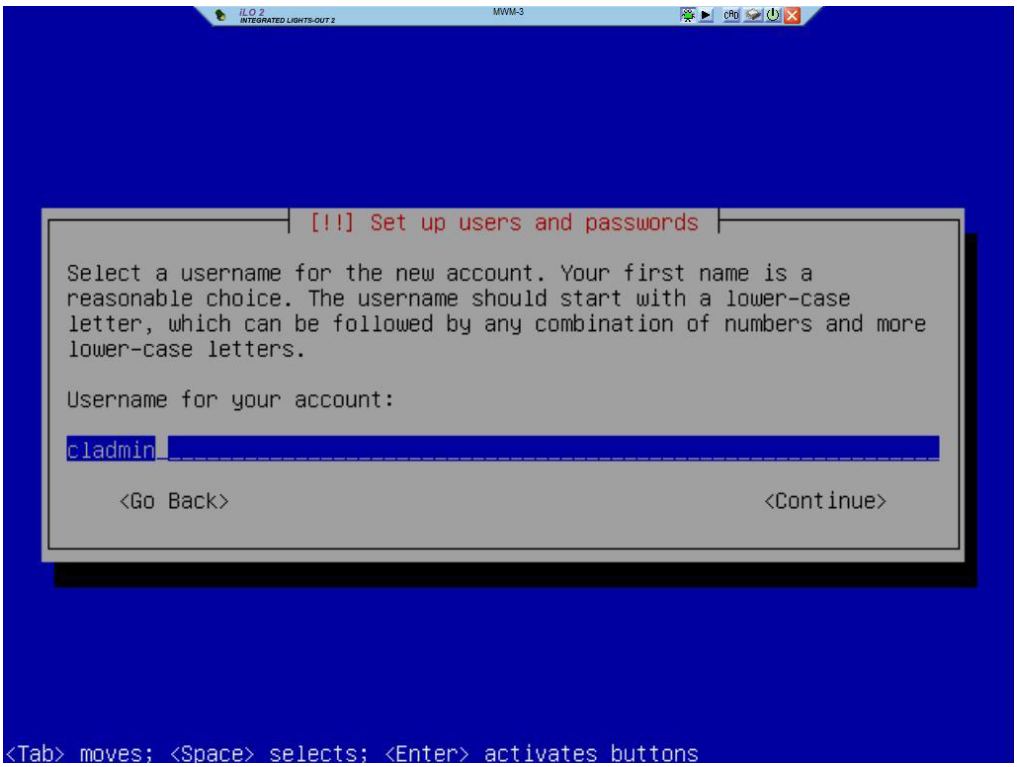
الشكل (A-12) حفظ التغيرات الجديدة على القرص الصلب

بعدها يتم تهيئة القرص الصلب وتنصيب نظام التشغيل كما هو مبين في الشاشة الشكل (A-13).



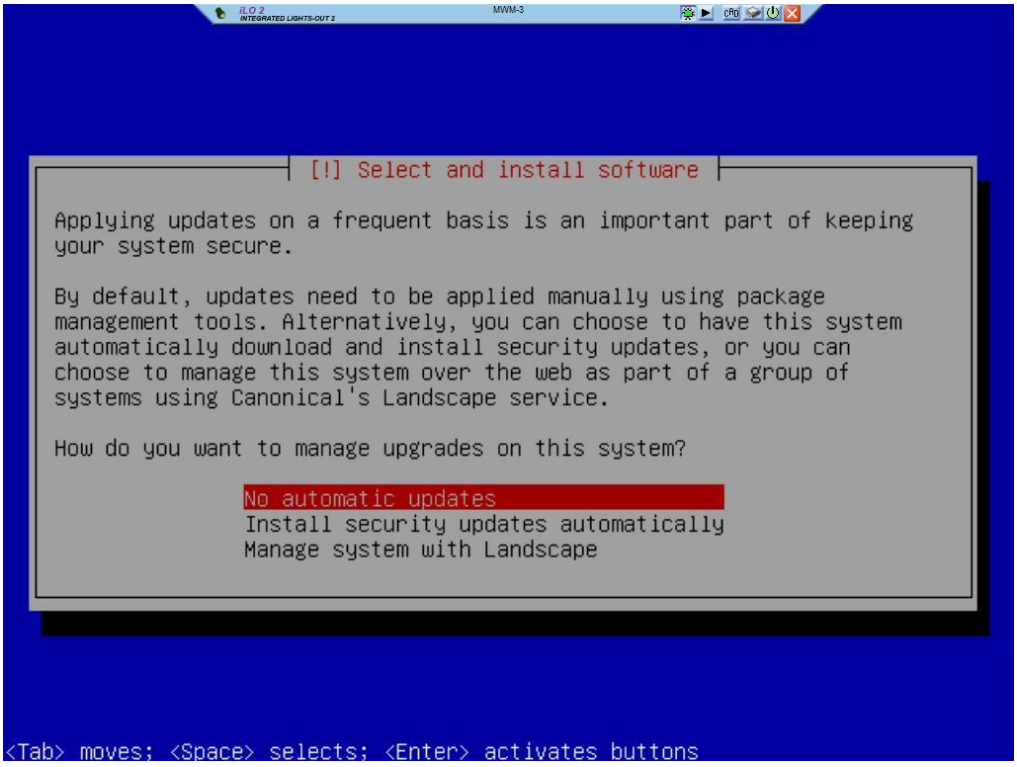
الشكل (A-13) تهيئة القرص الصلب وتنصيب نظام التشغيل

يتم هنا إدخال اسم المستخدم cladmin وكلمة المرور cloud9 كما هو مبين في الشاشة الشكل (A-14).



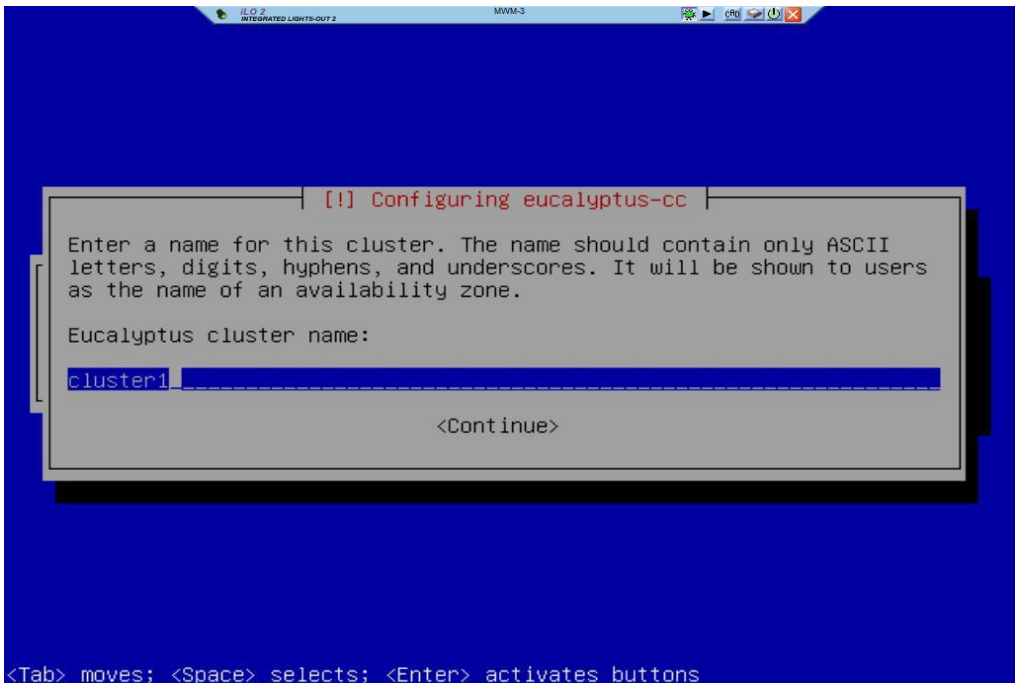
الشكل (A-14) إدخال اسم المستخدم cladmin وكلمة المرور cloud9

من الموصى عدم القبول بالتحديث هنا كما هو مبين في الشاشة الشكل (A-15).



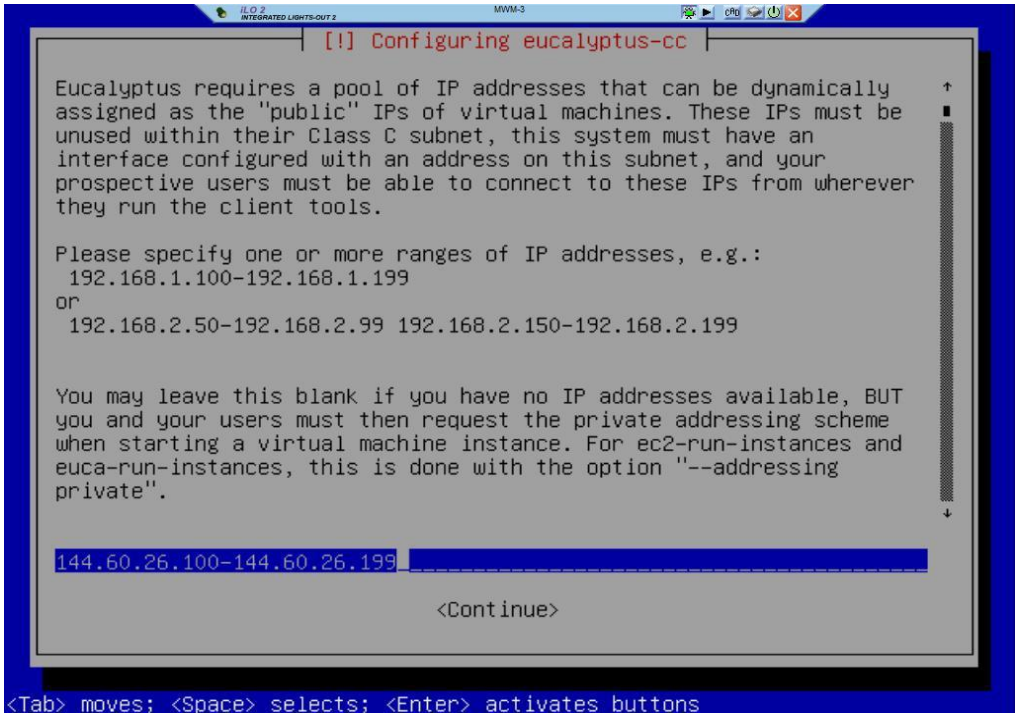
الشكل (A-15) عدم القبول بالتحديثات

بعد إدخال البريد الالكتروني، نكتب اسم العنقود كما هو مبين في الشاشة الشكل (A-16).



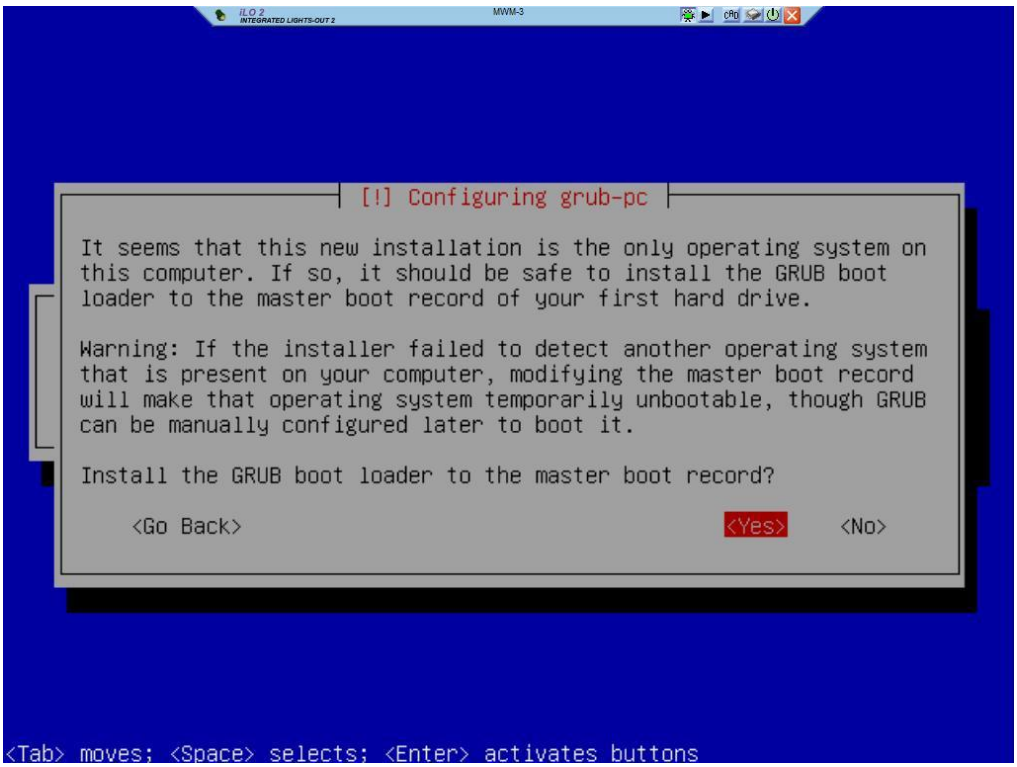
الشكل (A-16) كتابة اسم العنقود

ندخل هنا مجموعة من عنوان IP التي يتم إسنادها تلقائياً إلى مثيلات VMs، حيث يتم الوصول إليها من خارج السحابة كما هو مبين في الشاشة الشكل (A-17).



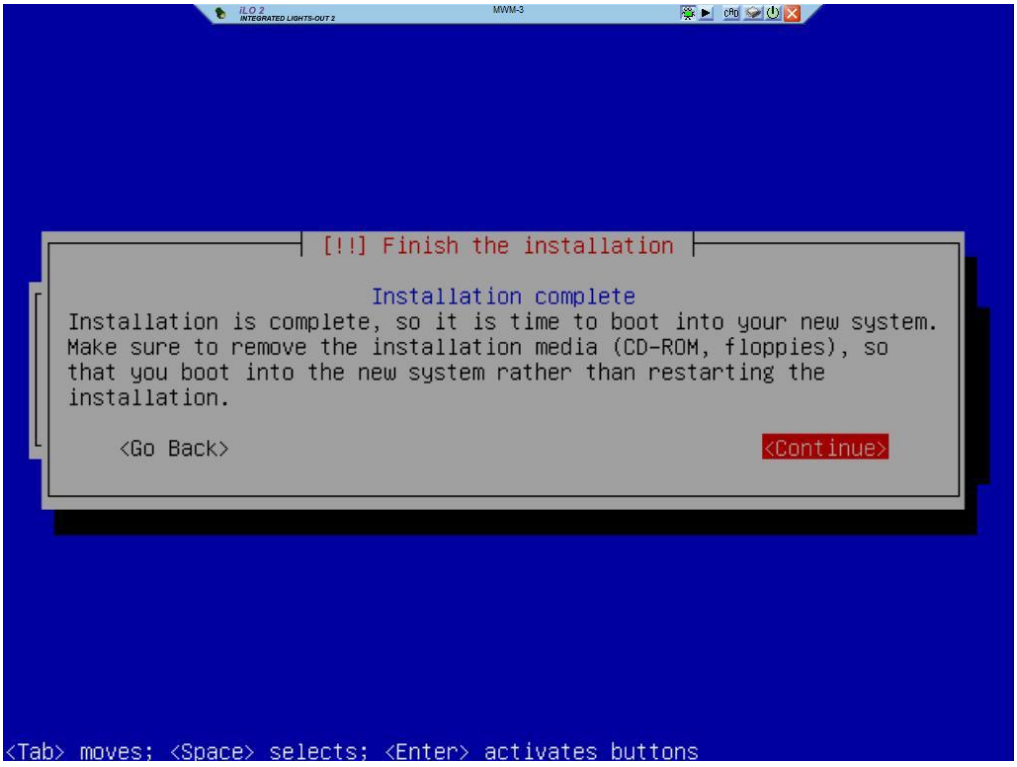
الشكل (A-17) إدخال مجموعة عناوين IPs

وفي الشاشة الشكل (A-18) سوف ننصب Grand Unified Boot loader, GRUB



الشكل (A-18) تنصيب GRUB

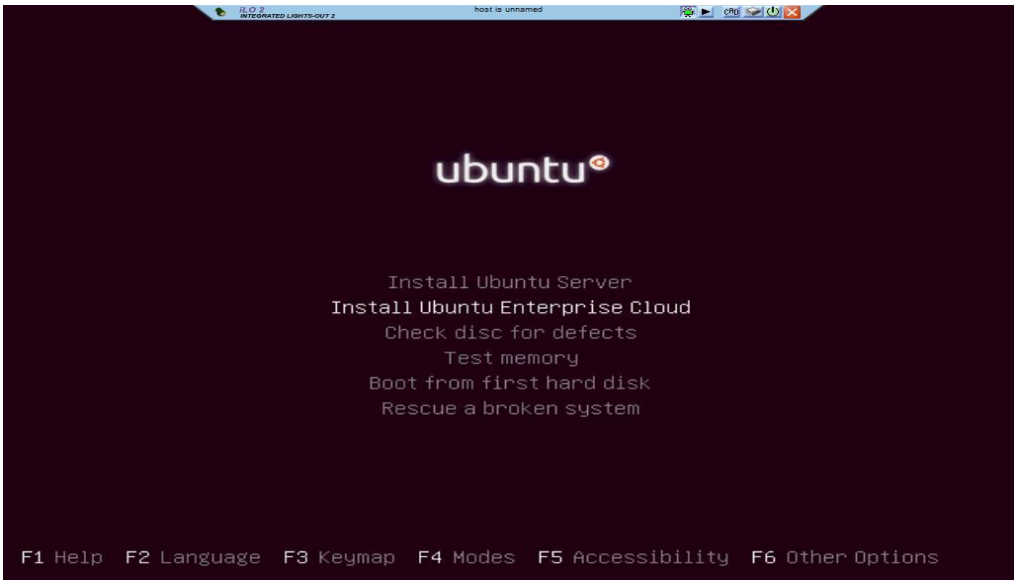
هنا يكون قد انتهى تنصيب متحكم السحابة كما في الشكل (A-19) .



الشكل(A-19) الانتهاء من تنصيب متحكم السحابة

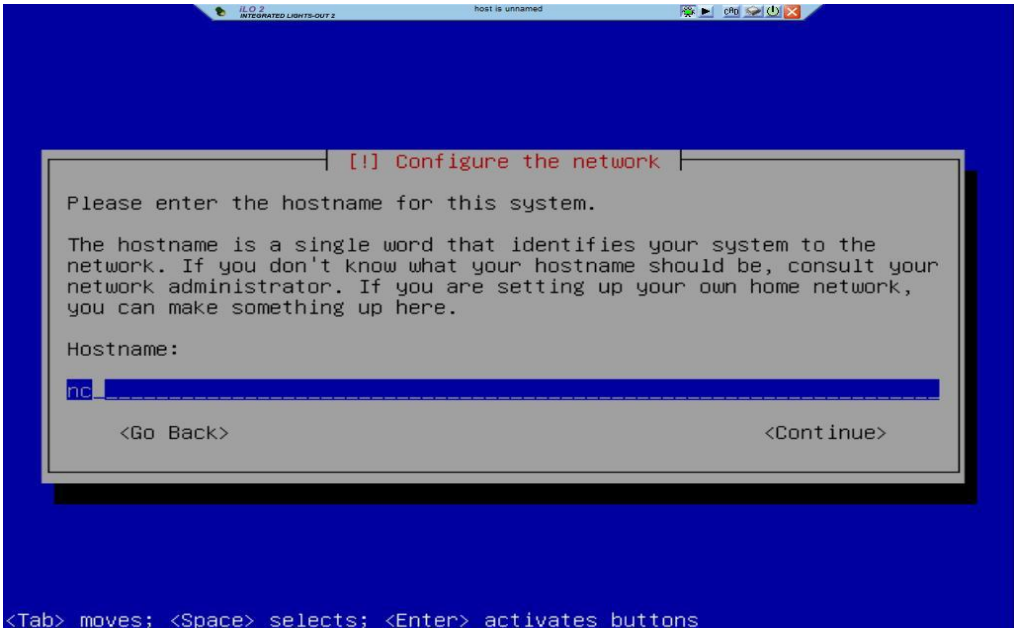
إنشاء متحكم العقدة:

حتى الآن قد تم إنشاء متحكم السحابة (متحكم العنقود، متحكم التخزين، walrus,)، بعدها ننتقل إلى المخدم الثاني لإنشاء متحكم العقدة من قرص Ubuntu 10.04 Server ونختار "Install Ubuntu Enterprise Cloud" من القائمة كما هو مبين في الشكل(A-20):



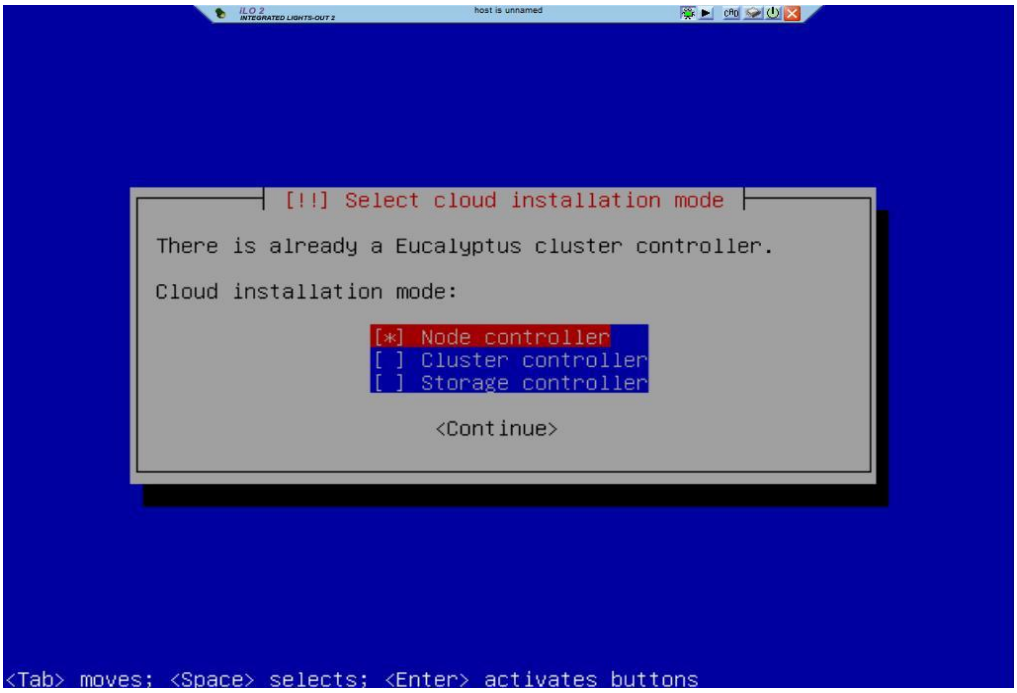
الشكل(A-20) تنصيب نظام تشغيل Ubuntu

وفي الشاشة الشكل(A-21) بعد اختيار اللغة المناسبة والبلد ولوحة المفاتيح وواجهة الشبكة، سوف ندخل اسم المضيف "nc" لاسم متحكم العقدة.



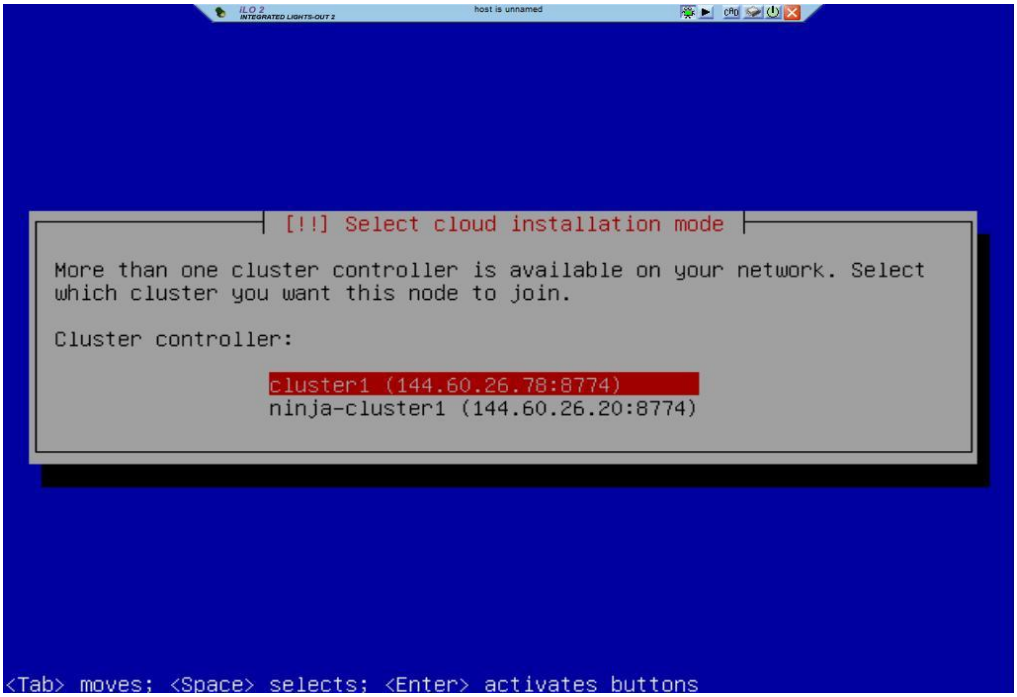
الشكل(A-21) إدخال اسم العنقود

عندها سوف يتم اكتشاف متحكم العنقود الذي يعمل على الشبكة، وسوف نختار من القائمة "Node Controller" كما في الشكل(A-22).



الشكل(A-22) اختيار متحكم العنقود

وبعد اختيار نمط تنصيب السحابة، عندها سوف نواجه شاشة مشابهة للشاشة إذا كان هناك متحكم عنقود في الشبكة الفرعية كما في الشكل(A-23).



الشكل(A-23) اختيار متحكم عنقود آخر موجود في الشبكة

تحذير: وقعنا في بعض القضايا عند تنصيب أكثر من سحابة واحدة على نفس الشبكة الفرعية، لذلك كن حذراً.

-في الخطوات التالية تواجهنا خطوات لتقسيم القرص الصلب، وبعدها سوف نستخدم نفس الإعدادات المستخدمة في متحكم السحابة، وبعدها يتم إعادة التشغيل.

الآن تم الانتهاء من تنصيب متحكم السحابة ومتحكم العقدة، والآن يمكن إدخال إعدادات الوصول الإداري للسحابة.

من الآن سوف نستخدم في سطر الأوامر الأسماء "cc" و "nc". وإذا لم يتم إعداد DNS في الشبكة، لذلك سوف نحتاج إلى تحديد IP بدلاً من اسم المضيف.

-إعدادات الدخول لمستخدم الـ Eucalyptus:

ملاحظة: لسنا بحاجة إلى هذه الخطوات إذا تم اكتشاف متحكم العقدة أثناء تنصيب متحكم السحابة.

الخطوة1: ندخل كلمة المرور بشكل مؤقت لحساب eucalyptus.

-الدخول إلى متحكم العقدة باستخدام user: "cladmin" password: "cloud9"

```
cladmin@nc:~$ sudo passwd eucalyptus
```

أدخل كلمة المرور "cloud9" بشكل مؤقت.

الخطوة2:

هنا يتم الدخول إلى متحكم السحابة ونسخ المفتاح العمومي للـ SSH لمستخدم eucalyptus إلى متحكم العقدة.

```
cladmin@cc:~$ sudo -u eucalyptus ssh-copy-id -i ~eucalyptus/.ssh/id_rsa.pub
eucalyptus@nc
```

الخطوة3:

الآن، سوف نزيل كلمة المرور المؤقتة من متحكم العقدة

```
cladmin@nc:~$ sudo passwd -d eucalyptus
```

-تنصيب الاعتمادية الإدارية للسحابة من خلال واجهة ويب Eucalyptus.

قبل استخدام سطر الأوامر Amazon EC2 للتعامل مع السحابة، نحتاج إلى تنصيب الاعتمادية التي تحتوي شهادة x.509 ومتغيرات البيئة.

الخطوة1:

استعرض الرابط https://cc:8443

الدخول باستخدام اسم المستخدم الافتراضي وكلمة المرور admin, admin، كما هو مبين في الشكل(A-24).



الشكل(A-24) واجهة الدخول إلى السحابة

الخطوة 2:

تغيير كلمة المرور لحساب admin ووضع الإيميل، عندها عنوان IP مضيف السحابة يملئ بشكل أتوماتيكي كما هو مبين في الشكل(A-25).



الشكل(A-25) واجهة تغيير كلمة المرور

الخطوة 3: سوف نعمل على تحميل اعتمادنا، حيث يتم تحديد واجهة ويب Eucalyptus حالياً، وبعد استخدام اعدادات معظم الإدارة سوف تكون منفذة من سطر الأوامر باستخدام أدوات Amazon EC2، اسم الحزمة في نظام Ubuntu هو "euca2ools" ويركب بشكل ملائم على متحكم السحابة،

لذلك سوف نستخدم سطر أوامر متحكم السحابة لإدارة السحابة. لتحميل "الاعتمادية" نختار الـ tab المخصصة لها وننقر تحميل "Download"، كما هو مبين في الشكل(A-26).



الشكل(A-26) تحميل الاعتمادية

الخطوة4: نسخ الملفات المحملة euca2-admin-x509.zip إلى المجلد /home/cladmin على متحكم السحابة، يمكن استخدام scp, ftp, sftp أو أية طريقة مفضلة أخرى.

الخطوة 5:

الآن يمكن إنشاء مجلد مخفي على متحكم السحابة وفك ضغط الملف السابق إلى هذا المجلد.

```
cladmin@cc:~$ mkdir ~/.euca
cladmin@cc:~$ cd ~/.euca
cladmin@cc:~/.euca$ unzip ../euca2-admin-x509.zip
```

الخطوة 6:

وبسبب احتواء ملف الاعتمادية على معلومات الوصول الإدارية المسموحة إلى السحابة، يوصى بحذف الملف المضغوط وإعطاء الامتيازات إلى مجلد euca. ومحتوياته:

```
cladmin@cc:~/.euca$ rm ~/euca2-admin-x509.zip
cladmin@cc:~/.euca$ chmod 0700 ~/.euca
cladmin@cc:~/.euca$ chmod 0600 ~/.euca/*
```

الخطوة 7:

سوف نضيف السطر ملف ~/.bashrc في متحكم السحابة للتأكيد أن متغيرات البيئة الضرورية تكون مستخدمة أثناء الدخول:

```
cladmin@cc:~/.euca$ echo  ". ~/.euca/eucarc"  >> ~/.bashrc
```

الخطوة 8:

سوف نحدد مصدر الملف bashrc. للتأكيد على أن التغيرات أخذت مفعولها.

```
cladmin@cc:~/.euca$ source ~/.bashrc
```

يمكن تسجيل الخروج للتأكد من الإعدادات فعالة.

-تنصيب صور السحابة Installing Cloud Images

تعرض صفحة الـ image قائمة بكل الـ image مسجلة في السحابة، بحيث أن كل مثيل (حالة) أو VM تشتغل في هذه السحابة تستند على هذه الـ image. لا يوجد أي Image بشكل افتراضي بعد الانتهاء من تنصيب السحابة، لذلك يجب تنصيبها كما هو مبين في الشكل(A-27).

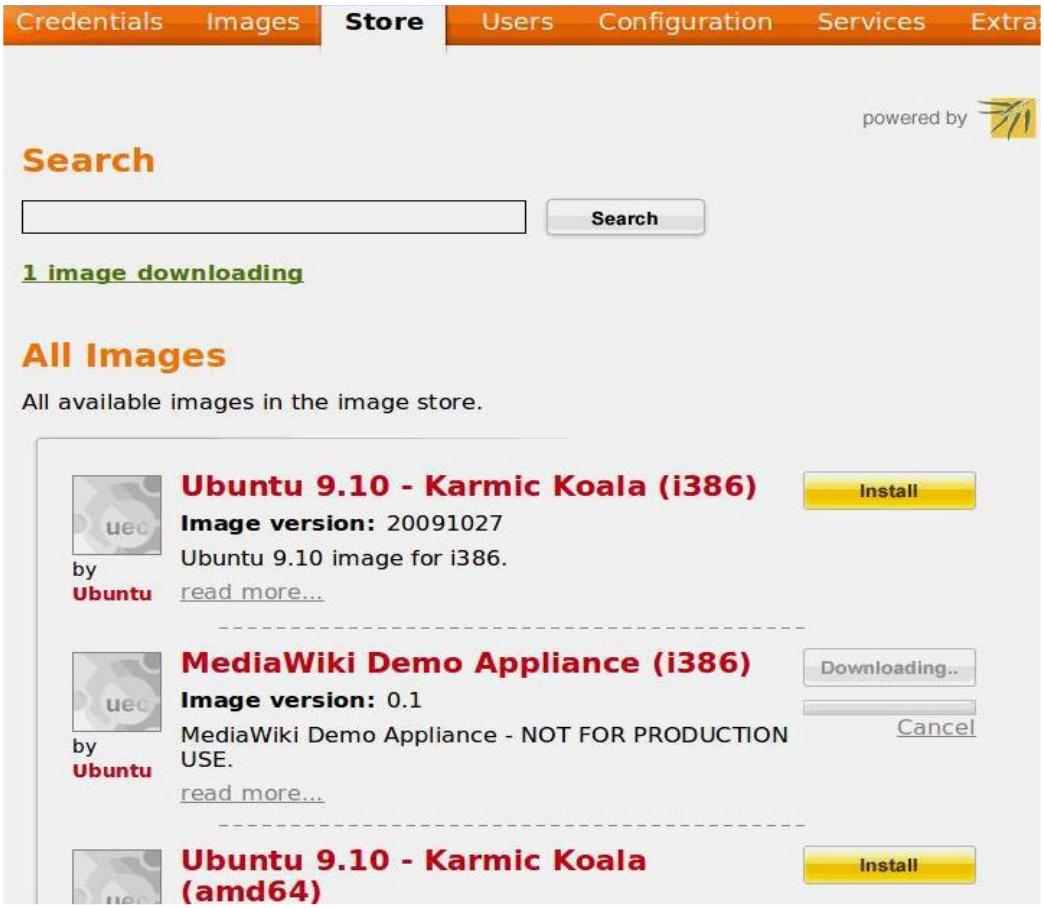


الشكل(A-27) صفحة Image

الخطوة 1:

حيث أنه يمكننا إنشاء ومعالجة الـ image الخاصة بنا، رفع وتسجيلهم مع السحابة، لذلك سوف نقوم بتنصيب هذه الـ image من الانترنت Canonical's online .

-انقر على صفحة الـ Tab "store " من واجهة الويب عندها سوف نشاهد الـ image المتوفرة من Canonical. من أجل تجربتنا سوف نقوم بتنصيب MediaWiki Demo Appliance image، وبعد تحميلها سوف نقوم بتنصيبها في السحابة كما هو مبين في الشكل(A-28).



الشكل(A-28) تنصيب الـ Image

الخطوة 2:

بعد تنصيب الـ image، نختار صفحة Images للتأكد أنها مسجلة ضمن السحابة كما هو مبين في الشكل(A-29).



الشكل(A-29) صفحة الـ Image

لاحظ أن القيمة emi-xxxxxx تحت عمود ID سيكون هو المعرف الذي نستخدمه لتشغيل المثيل (الحالة).

يكافئ ملف emi في سحابة Eucalyptus ملف Amazon Machine Image (AMI) من موقع آمازون. الخدمات التي تتألف من raw disk image والمؤشر إلى البنية وباختيار الـ ramdisk.

-تشغيل المثيل Instance:

قبل تشغيل المثيل يجب التأكد من أنه هناك موارد كافية متوفرة في السحابة (ومثال على ذلك العقد) نحن نستخدم euca-describe-availability-zones حتى نرى كافة الموارد المتوفرة في عقد السحابة.

الخطوة 1:التحقق من الموارد:

```
cladmin@cc:~$ euca-describe-availability-zones verbose
AVAILABILITYZONE cluster1 144.60.26.85
AVAILABILITYZONE |- vm types free / max cpu ram disk
AVAILABILITYZONE |- m1.small 0016 / 0016 1 192 2
AVAILABILITYZONE |- c1.medium 0016 / 0016 1 256 5
AVAILABILITYZONE |- m1.large 0008 / 0008 2 512 10
AVAILABILITYZONE |- m1.xlarge 0008 / 0008 2 1024 20
AVAILABILITYZONE |- c1.xlarge 0004 / 0004 4 2048 20
```

يمكن تعديل المناطق المتوفرة الافتراضية تحت صفحة المدير "Administration" في واجهة متصفح إدارة Eucalyptus.

الخطوة 2:

التحقق من الـ Images: الأمر "euca-describe-images" في سطر الأوامر يكافئ النقر على صفحة "Images" في واجهة متصفح إدارة Eucalyptus، وهذا يرينا معرف emi-xxxxxx لكل image/bundle الذي سوف يستخدم لتشغيل المثيل.

```
cladmin@cc:~$ euca-describe-images
IMAGE emi-E088107E image-store-1276733586/image.manifest.xml admin
available public x86_64machine eki-F6DD1103 eri-0B3E1166
IMAGE eri-0B3E1166 image-store-1276733586/ramdisk.manifest.xml admin
available public x86_64ramdisk
IMAGE eki-F6DD1103 image-store-1276733586/kernel.manifest.xml admin
available public x86_64kernel
```

الخطوة 3:

يتم وضع مجموع الأمان كقواعد في جدار النار iptable والتي تتحكم بطلبات الاتصال من المضيفين خارج السحابة وتوجيه تشغيل المثيلات الافتراضية في داخل السحابة.

يمكننا عرض مجموعات الأمان داخل Eucalyptus بإصدار الأمر التالي:

```
cladmin@cc:~$ euca-describe-groups
```

وذلك لأن مجموعات الأمان بشكل افتراضي "default" لا تحتوي أية قواعد تسمح بالخول من الخارج إلى مثيلات السحابة.

لذلك يمكن تعديل هذه القواعد في مجموعات الأمان أو إنشاء مجموعات جديدة واستخدامها بدلاً من المجموعات الافتراضية. في هذه التجربة سوف نستخدم الحروف "wiki" لإنشاء مجموعة جديدة.

```
cladmin@cc:~$ euca-add-group wiki -d wiki_demo_appliances
cladmin@cc:~$ euca-authorize wiki -P tcp -p 22 -s 0.0.0.0/0
cladmin@cc:~$ euca-authorize wiki -P tcp -p 80 -s 0.0.0.0/0
```

بتشغيل الأمر euca-describe-groups ثانيةً يمكننا من مشاهدة المجموعة الجديدة.

الخطوة 4: تركيب الـ Keypair

نحتاج لإنشاء Keypair التي سيتم حقنها في المثيل يسمح لنا بالدخول عن طريق ssh.

```
cladmin@cc:~$ euca-add-keypair mykey > ~/.euca/mykey.priv
cladmin@cc:~$ chmod 0600 ~/.euca/mykey.priv
```

الخطوة 5: تشغيل المثيل

لبدء تشغيل المثيلات (الحالة) سنبدأ من خلال إنشاء حالة تطبيق Mediawiki وتخصيص مجموعة أمان wiki وذلك للسماح بالاتصال مع منافذ SSH و http.

```
cladmin@cc:~$ euca-run-instances -g wiki -k mykey -t c1.medium emi-xxxxx
```

لاحظ بأنه إذا كانت المنطقة zone المتوفرة صغيرة المختارة من أجل الـ Image، سوف تنتهي بشكل أتوماتيكي بسبب عدم وجود فضاء العمل الكافي. اختبار الملف /var/log/eucalyptus/nc.log على العقدة يمكن أن يعطينا دلالة مفيدة في هذه الحالة.

مراقبة ودخول الحالات:

بعد إصدار أمر "euca-run-instances" لتشغيل الحالة، يمكن متابعة تقدمها من الانتظار إلى تشغيل الحالة باستخدام الأمر euca-describe-instances. يمكن أن نسجل ملاحظة عن عنوان الـ IP المخصص، لذلك يمكن أن نختبر دخول الحالة من خارج السحابة. يمكن تشغيل الأمر euca-run-instances بالارتباط مع خدمة "watch" لرؤية الناتج كل ثانية.

```
cladmin@cc:~$ watch -n1 euca-describe-instances
```

يمكن أن يكون مفيد في بعض الأوقات رؤية خرج الـ console للحالة. يمكن الحصول على خرج euca-get-console output باستخدام سطر الأوامر لهذه المهمة، حيث i-xxxxxx يقابل معرف ID الصورة image المدرجة من قبل "euca-describeinstances" الأمر:

```
cladmin@cc:~$ euca-get-console-output i-xxxxxxx
```

بسبب السماح لخدمة ssh في مجموعة الأمن، يمكن الدخول إلى wiki من خلال SSH باستخدام المفتاح الذي خصصناه عند إنشاء الحالة:

cladmin@cc:~$ ssh -i ~/.euca/mykey.priv ubuntu@w.x.y.z

باستخدام العنوان العام IP، يمكن استعراض العنوان للحالة للتأكد أن wiki متوفرة كما في الشكل (A-30).

http://w.x.y.z/mediawiki



الشكل (A-30) استعراض العنوان للحالة

للحصول على شعور بالأداء تحت الحمل، يمكن أن نسرع الحالة في المنطقة zone المتوفرة المتبقية.

نريد أن نتأكد إذا كانت متوفرة:

```
cladmin@cc:~$ euca-describe-availability zones verbose
AVAILABILITYZONE cluster1 144.60.26.85
AVAILABILITYZONE |- vm types free / max cpu ram disk
AVAILABILITYZONE |- m1.small 0015 / 0016 1 192 2
AVAILABILITYZONE |- c1.medium 0015 / 0016 1 256 5
AVAILABILITYZONE |- m1.large 0007 / 0008 2 512 10
AVAILABILITYZONE |- m1.xlarge 0007 / 0008 2 1024 20
AVAILABILITYZONE |- c1.xlarge 0003 / 0004 4 2048 20
```

يمكن أن نرى كم يأخذ لتسريع حالة الـ wiki image في DL380:

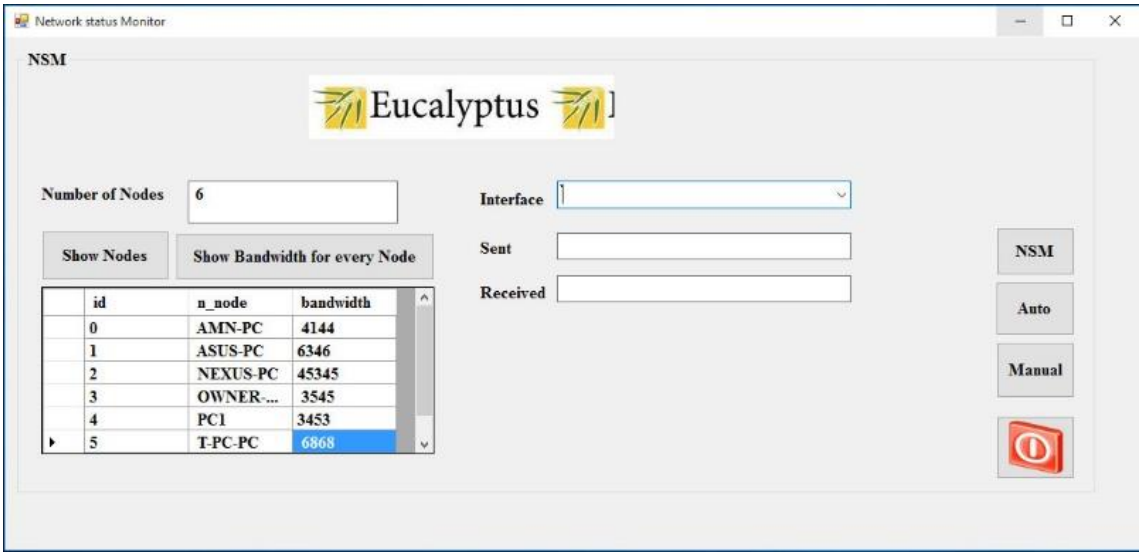
```
cladmin@cc:~$ euca-run-instances -g wiki -n xyz -k mykey -t c1.medium emi-xxxxx
cladmin@cc:~$ date
cladmin@cc:~$ watch -n2 euca-describe-instances
cladmin@cc:~$ date
```

ثانية يمكن زيارة الرابط لأية حالة لنرى إذا كانت تستجيب للاتصالات الخارجية.

الملحق –B–

مراقب حالة الشبكة Network Status Monitor

مراقب حالة الشبكة (Network Status Monitor [NSM]): هو برنامج مكتوب بلغة (C#) يحتوي على قاعدة بيانات تسجل العقد المتوفرة، وعرض الحزمة المستهلك من كل عقدة، ويتواجد في الحوسبة السحابية مراقب واحد. عندما يتلقى متحكم مراقبة حالة الشبكة (NSM) المضاف طلب من متحكم العنقود (CC)، يبدأ بمراقبة و تسجيل حالة استخدام الشبكة بين متحكم العنقود (CC) وكل متحكم عقدة ( )NC وتُعطى بالتسمية ((CN(i))، وبين المستخدمين ومتحكم العنقود (CC) وتُعطى بالتسمية (Cu-cc)، وبعد أن يحصل متحكم مراقبة حالة الشبكة (NSM) المضاف على جميع النتائج، يحسب السعة الوسطية ()w لجميع سعات اتصال متحكمات العقد المتوفرة CN(i). ثم يقوم متحكم مراقبة حالة الشبكة (NSM) عندها بالمقارنة بين (w،Cu-cc) ويختار متحكم العقدة (NC) الأنسب بناءً على الآلية المقترحة في الفصل الخامس. عند ذلك يقوم متحكم العنقود (CC) بتشغيل الآلة الافتراضية على متحكم العقدة الذي تم اختياره. ويبين الشكل (1) واجهة برنامج متحكم الشبكة حيث تم تنفيذ مقلد لهذا المراقب متوافق مع أنظمة التشغيل Windows وذلك من أجل توضيح آلية عمل هذا المراقب.



الشكل (B-1) واجهة برنامج مراقب حالة الشبكة

المراجع

[1] John McCarthy, speaking at the MIT Centennial in 1961, "Architects of the Information Society, Thirty-Five Years of the Laboratory for Computer Science at MIT," Edited by Hal Abelson.

[2] Supreeth S, Shobha Biradar, "Scheduling Virtual Machines for Load balancing in Cloud Computing Platform" International Journal of Science and Research (IJSR), India Online ISSN: 2319-7064, Volume 2 Issue 6, June 2013.

[3] G.Suryadevi, D.Vijayakumar, R.SabariMuthuKumar, Dr. K .G. Srinivasagan, "An Efficient Distributed Dynamic Load Balancing Algorithm for Private Cloud Environment", International Journal of Innovative Research in Science, Engineering and Technology Volume 3, Special Issue 3, March 2014.

[4] Jiann-Liang Chen, Yanuarius Teofilus Larosa and Pei-Jia Yang, "Optimal QoS Load Balancing Mechanism for Virtual Machines Scheduling in Eucalyptus Cloud Computing Platform", 978-1-4673-1671-2/12/$31.00 c_ 2012 IEEE

[5] 1. J. Voas and J. Zhang, "Cloud Computing: New Wine or Just a New Bottle?" IEEE ITPro, March/April 2009, pp. 15-17.

[6] Peter Mell, Timothy Grance, Recommendations of the National Institute of Standards and Technology "The NIST Definition of Cloud Computing"

[7] F. Gens, "Defining Cloud Services and Cloud Computing," September 2008. http://blogs.idc.com/ie/?p=190

[8] . R. L. Grossman, "The Case for Cloud Computing," IEEE ITPro, March/April 2009, pp. 23-27.

[9] R. L. Grossman, "The Case for Cloud Computing," IEEE ITPro, March/April 2009, pp. 23-27.

[10] "Which Cloud Platform is Right for You?" www.cumulux.com. Last visit 12/2/2016.

[11] G. Lakshmanan, "Cloud Computing – Relevance to Enterprise," Infosys White Paper, April 2009.

[12] G. Lakshmanan, "Cloud Computing – Relevance to Enterprise," Infosys White Paper, April 2009.

[13] "Introduction to Cloud Computing Architecture," White Paper, Sun Microsystems, June 2009.

[14] G. Lakshmanan, "Cloud Computing – Relevance to Enterprise," Infosys White Paper, April 2009.

[15] N. Ajith Singh, M. Hemalatha, "An approach on semi distributed load balancing algorithm for cloud computing systems", International Journal of Computer Applications Vol-56 No.12 2012.

[16] N.S. Raghava and Deepti Singh, "Comparative study on Load balancing techniques in cloud computing", Open Journal of Mobile Computing and Cloud Computing. Vol. 1, Num. 1 . August 2014. PP 18-25.

[17] Ruhi Gupta, "Review on Existing Load Balancing Techniques of Cloud Computing", International Journal of Advanced Research in Computer Science and Software Engg. Vol.4 Iss. 2, Feb. 2014. Pp: 168-171

[18]http://www.writeaprogram.info/c/os-programs/priority-scheduling/"Priority Scheduling Algorithm Example – Write a Program.

[19] R. Alonso-Calvo, J. Crespo, M. Garcia-Remesal, A. Anguita, V. Maojo, "On distributing load in cloud computing:A real application for very-large image datasets", Procedia Computer Science 1 (1) (2010) pages 2669–2677.
[20] Zenon Chaczko, Venkatesh Mahadevan, Shahrzad Aslanazadeh, and Christopher, IPCSIVol-14, IACSIT Press Singapore 2011.
[21] Tushar Desai, Jignesh Prajapati, "A Survey of Various Load Balancing Techniques and Challenges in Cloud Computing", International Journal of Scientific & Technology Research. Vol.2 Iss.11. Nov.2013 PP:158-161.
[22] Shanti Swaroop Moharana1, Rajadeepan D. Ramesh2 & Digamber Powar, "Analysis Of Load Balancers In Cloud Computing"
[23] Graham Ritchie, John Levine, "A fast effective local search for scheduling independent jobs in heterogeneous computing environments", Center for Intelligent Systems and their applications School of Informatics University of Edinburg.
[24] O M Elzeki, M Z Reshad," Improved Max-Min Algorithm in Cloud Computing, International Journal of Computer Applications". Vol. 50, Iss. 12. July 2012. PP: 22-27.
[25] Graham Ritchie, John Levine, "A fast effective local search for scheduling independent jobs in heterogeneous computing environments" Center for Intelligent Systems and their applications School of Informatics University of Edinburg.
[26] Che-Lun Hung, Hsiao-hsi Wang, and Yu- ChenHu "Efficient Load balancing Algorithm for cloud computing network"
[27] Nitika, Shaveta, Gaurav Raj, International Journal of advanced research in computer engineering and technology, Vol-1 issue-3 May-2012.
[28] Nidhi Jain Kansal, Inderveer Chana , "Existing Load balancing Techniques in cloud computing: A systematic review", Journal of Information system and communication Vol-3 Issue-1 2012
[29] Nayandeep Sran, Navdeep kaur "Comparative Analysis of Existing Load balancing techniques in cloud computing", International Journal of Engineering Science Invention, Vol-2 Issue-1 2013
[30] *Graziano, Charles.* "A performance analysis of Xen and KVM hypervisors for hosting the Xen Worlds Project". Retrieved 2013-01-29.
[31] "Virtualization in education" *(PDF).* IBM. *October 2007*. Retrieved 6 July 2010.
[32] *Creasy, R.J. (1981).* "The Origin of the VM/370 Time-sharing System" *(PDF).* IBM. Retrieved 26 February 2013.
[33] *VMware (11 Sep 2007).* "Understanding Full Virtualization, Paravirtualization, and Hardware Assist" *(PDF).* VMware. Retrieved 2007-12-09.
[34] " https://software.intel.com/sites/common/403_compliance.htm" last visit at 22/4/2016.
[35] 5. V. Amiry, S.Z. Rad, M.K. Akbari, and M.S. Javan, "Implementing Hadoop Platform on Eucalyptus Cloud Infrastructure," Proceedings of the 3PGCIC eventh International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, pp.74-78, November 2012.
[36]" https://academy.hsoub.com/devops/web-servers/nginx" last visit at 22/4/2016.