

УДК 378.16+004.27+004.9+004.75+004.052.3+004.056.4

Лозінський А. П.

ОГЛЯД ФУНКЦІОНАЛЬНИХ МОЖЛИВОСТЕЙ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ХМАРНОЇ ПЛАТФОРМИ OPENSTACK ICEHOUSE

У статті висвітлена інформація про функціональні можливості проекту хмарної обчислювальної платформи OpenStack Icehouse з відкритим програмним кодом. Наведено перелік програмних служб, які формують архітектуру та функціональні можливості платформи OpenStack Icehouse. Зокрема висвітлюються функціональні можливості оркестрації і механізми забезпечення високої доступності та надійності хмарної платформи, методи забезпечення безпеки хмарних обчислень.

Ключові слова: розподілена обчислювальна система, хмарна обчислювальна платформа, оркестрація, динамічна архітектура, віртуальні машини, висока доступність, надійність, безпека хмарних обчислень.

Бурхливий розвиток інформаційних технологій (ІТ) та технологій обчислювальної техніки призвів до появи нового напрямку – хмарних обчислень. У науковій літературі висвітлюються можливості постачальників використання послуг хмарних обчислень прикладного спрямування (праці L. Badger, T. Grance, R. Patt-Corner, J. Voas, W. Jansen, T. Grance та інших). Актуальність теми зумовлена вагомих місцем яке займають технології хмарних обчислень як у академічній, так і у виробничій галузі. Сприятливі умови для такого плину подій створила довіра до технологій використання віртуальних машин (VM). За даними аналітичної компанії IDC у 2010 році кількість інсталяцій операційних систем на VM перевищила кількість інсталяцій на фізичні машини, а в 2013 році вже становила дві треті загального обсягу [1]. Інтерес до хмарних інформаційних систем постійно зростає у всіх сферах діяльності суспільства. Сучасні друковані видання та інформаційні ресурси в Інтернет постійно висвітлюють різні ідеї застосування хмарних обчислень та висловлюють позитивні відгуки про цілу низку можливостей, які відкриваються перед користувачами хмарних обчислювальних систем (ХОС) [2, 3, 4, 5]. У якості ключових ідей, що принципово відрізняють хмарні обчислення від традиційних підходів, проголошуються такі якості ХОС як теоретично необмежене масштабування, висока доступність та надійність. Проте слід зазначити, що принципово неможливо віднести такі твердження до будь-якої ХОС [6]. Насправді, залежно від сценарію за котрим будується ХОС та переліку технологій, які закладаються у процес її побудови будуть залежати її властивості. Використання ХОС користувачами у якості послуги, породжують цілу низку неоднозначних питань. Так на разі, якщо ХОС реалізована локально на території підприємства користувача, він може

розраховувати на її високу доступність. Якщо ж користувач використовує для підключення до ХОС канали зв'язку та комп'ютерні мережі, працездатність яких він не може контролювати, наприклад Інтернет, забезпечення високої доступності стає неможливим. Станом на 2014 рік створено велику кількість постачальників послуг хмарних обчислень різного прикладного спрямування. Постачальники послуг не розкривають способу реалізації їхніх ХОС, вважаючи цю інформацію приватною інтелектуальною власністю. Тому для огляду функціональних можливостей ХОС було обрано проект з відкритим програмним кодом [7] OpenStack Icehouse.

1. Загальні відомості про проект хмарної платформи OpenStack Icehouse.

Проект OpenStack Icehouse являє собою хмарну обчислювальну платформу IaaS (інфраструктура як послуга), яка підтримує широкий спектр функціональних можливостей хмарних обчислювальних систем. Користувачам надається послуга доступу до обчислювальних ресурсів хмарної платформи. Під обчислювальними ресурсами мається на увазі апаратні потужності обчислювальних машин для обробки та зберігання даних з дистанційним доступом. Створене в рамках проекту програмне забезпечення може бути використане як для створення її користувачами невеликих малопотужних інформаційних систем, так і для створення потужних, крупно-масштабних інформаційних систем з широким набором функцій.

У якості програмної платформи, на базі якої ведеться розробка OpenStack використовується операційна система Linux. Розробка та розповсюдження програмного забезпечення проекту OpenStack виконується за ліцензією Apache 2.0. Ліцензія Apache 2.0 являється класичною ліцензією для розповсюдження програмного забезпечення в проектах з відкритим програмним кодом. Ця ліцензія не забороняє змінювати умови ліцензування а також не примушує зберігати відкритість коду та безкоштовність програмного забезпечення. Вихідні тексти програм створених під цією ліцензією, або похідні від нього, можуть стати закритими та розроблятися на комерційній основі.

2. Архітектура сервісів хмарної платформи.

Проект програмної обчислювальної платформи OpenStack Icehouse організований таким чином, щоб забезпечити надання доступу до обчислювальних ресурсів за моделлю "Інфраструктура як послуга" (IaaS) шляхом використання набору взаємопов'язаних програмних служб. Кожна служба забезпечена програмним інтерфейсом (API), використання яких в результаті надає можливість інтеграції. Нижче наведено перелік служб, котрі формують архітектуру OpenStack [8]:

1. Веб-інтерфейс самообслуговування Horizon. Реалізує веб-інтерфейс порталу самообслуговування. Надає можливість адміністраторам та користувачам ХОС виконувати задачі контролю та

управління ресурсами ХОС використовуючи графічний інтерфейс.

2. Обчислювальна служба Nova. Керує життєвим циклом обчислювальних екземплярів ВМ. Виконує по запиті користувачів створення, запуск, зупинку та знищення ВМ, відповідно до задачі виділяючи або вивільняючи такі обчислювальні ресурси ХОС як процесорні ядра та оперативна пам'ять.

3. Мережа Neutron. Виконує роль забезпечення мережевої взаємодії між ВМ. По запиті користувачів ХОС служба Neutron створює, змінює та знищує програмно-конфігуровані мережі (англ. Software Defined Networks, SDN). SDN формуються шляхом створення та знищення, програмних комутатори та маршрутизаторів, включення апаратних інтерфейсів до складу програмних комутаторів, створення логічних мережевих з'єднань між програмними маршрутизаторами, комутаторами та екземплярами ВМ.

4. Блочне сховище Cinder. Служить для забезпечення обчислювальних екземплярів віртуальними блочними пристроями збереження інформації. Забезпечує можливість створення, знищення, управління та підключення до обчислювальних екземплярів блочних пристроїв збереження інформації.

5. Служба збереження образів Glance. Виконує функції колекції системних томів на кожному з яких знаходиться образ встановленої операційної системи. Служба використовується в процесі створення обчислювальних екземплярів. Згідно вибору користувачем необхідної для вирішення завдання операційної системи, завантажується відповідний образ системного тому операційної системи з колекції Glance.

6. Об'єктне сховище Swift. Виконує функції зберігання та надання неструктурованих об'єктів даних з використаннями API та протоколу HTTP. Програмне забезпечення служби Swift дозволяє реалізовувати об'єктні сховища даних витривалих до збоїв та надає можливість реалізувати широкі рамки об'ємів масштабування сховища.

7. Служба ідентифікації та авторизації Keystone. Забезпечує функції ідентифікації та авторизації користувачів, адміністраторів та служб OpenStack у процесі взаємодії.

8. Служба оркестрації Heat. Реалізує можливість управління всіма компонентами платформи OpenStack за допомогою програмних шаблонів написаних на мові HOT (англ. Heat Orchestration Template).

9. Телеметрія Ceilometer. Виконує функції моніторингу та обліку використання користувачами обчислювальних ресурсів ХОС. Облік ведеться в рамках створеного для користувача проекту за кількістю та часом використання процесорних ядер, оперативної пам'яті, простору сховища даних, швидкості та кількості переданих через мережу даних.

10. Служба баз даних Trove (СУБД як сервіс). Надає користувачам ХОС можливість користування базами даних СУБД MySQL для функціонування котрих закладено такі можливості як захист від збоїв та

масштабування. В наступних версіях OpenStack передбачається підтримка сервісом Trove СУБД PostgreSQL.

Комплексне використання користувачами хмарної платформи OpenStack перелічених вище служб надає можливість створювати VM, запускати в них широкий спектр операційних систем, встановлювати в операційні системи та запускати широкий перелік програмного забезпечення.

Користувач хмарної платформи OpenStack, після отримання облікового запису доступу, отримує можливість самостійно, без участі технічного персоналу, створювати свою інфраструктуру обчислювальних ресурсів в рамках заданих попередніми домовленостями обмежень на кількість доступних в рамках проекту ресурсів.

Закладений в архітектуру програмного забезпечення OpenStack принцип модульності дозволяє об'єднувати під його управлінням теоретично необмежену кількість апаратних обчислювальних ресурсів. Це можуть бути великого розміру сховища даних, з великою кількістю обчислювальних вузлів, процесорних ядер та оперативної пам'яті.

3. Прикладне використання хмарної платформи користувачами.

В першу чергу користувачі платформи мають можливість отримувати доступ до обчислювальних ресурсів шляхом самостійного створення VM. Користувач визначає кількість обчислювальних ресурсів котрі мають бути задіяні для забезпечення необхідної потужності VM котру він створює, а саме: кількість процесорних ядер, кількість оперативної пам'яті, величину накопичувача інформації та кількість мережевих інтерфейсів.

Користувач вказує не тільки параметри обчислювальних потужностей, але й операційну систему яку він бажає запустити на створеній VM. У хмарній платформі OpenStack передбачена можливість завантаження користувачами файлів образів встановлених та можливо сконфігурованих для конкретних задач операційних систем. Функція завантаження та зберігання образів операційних систем забезпечується службою Glance. У подальшій роботі користувач вибирає у процесі створення VM образ, з котрого має завантажуватися операційна система у VM[9].

Окрім виділення обчислювальних ресурсів для створення окремих VM, користувачі хмарної платформи мають можливість створювати та знищувати цілі віртуальні інфраструктури з'єднаних між собою VM зі встановленням на них, відповідно до вирішеної задачі, програмне забезпечення за допомогою служби оркестрації Heat та спеціалізованої програмної мови написання шаблонів оркестрації Heat Orchestration Template (HOT шаблон).

Таким чином, у разі необхідності, застосування певних обчислювальних потужностей та програмного забезпечення для вирішення користувачем певної задачі, користувач має можливість завантажити у проект попередньо створений HOT шаблон у службу Heat. У результаті

обробки шаблону службою Heat буде створено необхідна кількість VM із заданим у шаблоні програмним забезпеченням та алгоритмів їх взаємодії між собою, необхідних для вирішення поставленої користувачем задачі.

По закінченню вирішення задачі користувач знищує VM, чим вивільняє обчислювальний ресурс для використання іншими користувачами хмарної платформи. Таким чином можна збільшити ефективність використання обчислювального обладнання, зменшуючи час його простою. На апаратному обладнанні можуть тимчасово створюватися та після використання знищуватись різні інформаційні системи, орієнтовані на вирішення різних завдань.

Кожна створена користувачем інформаційна система буде мати динамічну архітектуру. Користувачі, при створенні своїх інформаційних систем у рамках хмари під керуванням OpenStack, можуть масштабувати кількість задіяних обчислювальних ресурсів в залежності від потреб вирішуваної задачі. Так наприклад, у разі вирішення ресурсоємкої задачі користувач створює інформаційну систему, у якій задіяна велика кількість VM із великою кількістю процесорних ядер, оперативної пам'яті та дискового простору. Ту ж саму інформаційну систему користувачі можуть створювати з меншими об'ємами задіяних ресурсів, якщо наступного разу передбачається обробка меншої кількості даних.

4. Реалізація високої доступності та надійності.

Реалізація високої доступності та надійності має мету зменшити вплив двох факторів. Перший, це час простою системи, котрий призводить до недоступності системи користувачам протягом максимально допустимого проміжку часу. Другий фактор, це втрата даних, а саме прецеденти знищення або руйнування даних.

Ключовим фактором реалізації високої доступності являється виключення існування єдиної точки збою системи. Єдиною точкою відмови вважається окремий програмний або апаратний елемент системи, у випадку збою якого відбувається втрата даних або збій всієї системи. Усунення єдиної точки збоїв реалізується шляхом забезпечення механізмів резервування мережевих компонентів, програм та можливості їх міграції, пристроїв збереження інформації, компонентів електричного живлення, повітряного охолодження, пожежного захисту. Для забезпечення механізмів високої доступності хмарної платформи OpenStack застосовується декілька методів.

Першим методом являється створення програмного кластера на базі взаємодії програмного забезпечення кластерних служб Corosync та Pacemaker [10]. Інструментарій кластерного програмного забезпечення дозволяє реалізувати дублювання служб OpenStack, що дозволяє вирішити завдання забезпечення високої доступності. У разі виходу з ладу апаратного або програмного забезпечення на вузлах, що входять до складу хмарної платформи, робоче навантаження бере на себе

дублюючий екземпляр служби на іншому апаратному вузлі. У наведеному методі величина робочого навантаження обмежена обчислювальним ресурсом тільки одного апаратного вузла, того на якому виконується активний екземпляр служби. Для збільшення обчислювальної потужності використовується другий метод.

Другий метод також використовує принцип побудови програмного кластера, який об'єднує потужності декількох апаратних вузлів для обробки робочих навантажень. Але, на відміну від першого методу, кластер будується таким чином, щоб робоче навантаження рівномірно розподілялося між службами, що виконують аналогічну роботу на декількох апаратних вузлах. У разі вилучення одного або декількох апаратних вузлів, робоче навантаження обробляють екземпляри служб на тих вузлах, що залишились працювати у складі хмарної платформи. Базовими службами, котрі використовуються для обміну інформацією між службами використовується служба RabbitMQ, а для зберігання робочої інформації СУБД MySQL. Для запуску MySQL у режимі кластера використовується програмне забезпечення Galera. Служба RabbitMQ має влаштовані можливості для запуску в режимі кластера. Для балансування навантаження між API службами OpenStack використовується програмне забезпечення Nagroху.

Наявність кількох методів забезпечення високої доступності дозволяє використовувати, в залежності від виду задачі той метод, котрий забезпечує кращий результат.

5. Безпека хмарних обчислень.

Закладений в ідею хмарних обчислень принцип самостійної роботи користувача з задачами, котрі в традиційних інформаційних системах вимагають участі технічного персоналу, породжує необхідність обмеження можливостей користувача в рамках достатніх для отримання ним очікуваної послуги. У першу чергу ця умова виникає з огляду на одночасне використання хмарної платформи великою кількістю користувачів. Програмне забезпечення хмарної платформи повинно забезпечити ізоляцію створених різними користувачами інформаційних систем одну від одної. Тип надання послуги програмним забезпеченням OpenStack відноситься до категорії "Інфраструктура як послуга", а це означає, що ізоляція виділених користувачам обчислювальних ресурсів між собою виконується на рівні керування гіпервізорами, під управлінням яких створюються VM та програмні мережі. Користувачі хмарної платформи не мають доступу до керування апаратним забезпеченням та операційними системами на базі котрих побудована хмарна платформа OpenStack. Проте, користувачі мають доступ до широкого спектру команд керування працюючих на їх базі гіпервізорів та архітектури служб OpenStack.

Для кожного користувача хмарної системи OpenStack заводиться облікових запис. Зберігання облікових записів виконується службою

Keystone. Крім облікових записів користувачів у базі служби Keystone ведеться облік об'єктів за допомогою яких забезпечується контроль рівня доступу та права доступу користувачів. Основним об'єктом за допомогою якого контролюється доступ користувача являється об'єкт типу "Проект". Всі створені користувачем хмари об'єкти (тобто, VM, пристрої зберігання інформації, мережеві пристрої і т.п.) групуються у рамках проекту, доступ до якого надається користувачу. Доступ до створених об'єктів в рамках окремого проекту за замовчуванням має тільки той користувач, котрий їх створив. Проте користувач має можливість змінити властивості доступу до об'єктів надаючи доступ до нього іншим користувачам хмари.

Для спрощення процедури надання доступу до проектів існує тип об'єкту "Група користувачів" котрий застосовується для об'єднання користувачів у типові групи та подальшого використання в процедурі надання доступу для цих груп.

У систему виділення закладено алгоритм контролю максимальної кількості ресурсів, доступної у рамках одного проекту. Таким чином, користувач має право на використання апаратних ресурсів хмарної системи у межах заданих адміністратором, що виключає можливість конфліктної ситуації використання одним користувачем всіх ресурсів хмарної платформи завдаючи такими діями шкоди іншим користувачам.

У той час, як метод ізоляції ресурсів користувачів один від одного, вирішує завдання безпеки доступу до конфіденційної інформації між користувачами, цей метод не є достатнім для захисту інформації від персоналу, котрий обслуговує апаратне та програмне забезпечення хмарної платформи. Для забезпечення захисту конфіденційних даних, у цьому випадку, необхідно застосовувати надійні алгоритми шифрування даних. Необхідно враховувати необхідність дешифрування даних для їх обробки та шифрування для подальшого зберігання. Необхідно застосовувати шифрування також у разі необхідності забезпечення конфіденційності даних, які передаються по мережі, як в межах так і за межами хмарної платформи. Рішення цього питання полягає у застосуванні встановлених у VM засобів відповідного програмного забезпечення та його налаштування.

Реалізація контролю мережевого доступу, як до операційних систем на базі яких побудована хмарна платформа OpenStack, так і ізоляція мережевих інфраструктур користувачів реалізується за допомогою правил програмного забезпечення служби мережевого екрану iptables, розбиття віртуальної та апаратної мережі на сегменти за допомогою мережевої технології VLAN та об'єднання сегментів за допомогою тунелів GRE.

Користувач має можливість створювати правила мережевого екрана фільтрації мережевих пакетів, що надходять у створену ним інформаційну систему. На базовому рівні мережевий екран являє собою низку правил iptables операційних систем Linux на базі котрих розгортається хмарна

платформа OpenStack. Користувач хмарної платформи не має доступу до безпосереднього керування правилами служби iptables операційної системи Linux, а керує тільки окремим набором правил таблиці iptables, який фільтрує мережевий трафік тільки для створеної ним інфраструктури інформаційної системи.

Питання безпеки хмарних обчислень включають ті самі аспекти, що притаманні традиційним обчислювальним системам [11, 12]. Застосування віртуалізації та можливості одночасного використання обчислювального обладнання великою кількістю користувачів збільшує ризик вразливості для атак хмарної платформи. Детальний аналіз конкретних сценаріїв реалізацій хмарних платформ на базі OpenStack повинен забезпечувати виявлення вразливих для атак факторів та реалізовувати відповідні засоби безпеки з метою їх усунення або зменшення ризиків.

Отже, програмне забезпечення проекту OpenStack являє собою перспективну високотехнологічну розробку в галузі інформаційних технологій, яка забезпечує можливість оперативного створення хмарних обчислювальних систем з можливістю масштабування задіяних у них обчислювальних ресурсів. Застосування хмарної платформи надає можливість гнучкого та ефективного використання апаратних ресурсів завдяки можливості швидкого створення тимчасової віртуальної інфраструктури з допомогою шаблонів оркестрації HОT.

Можливість вивільнення апаратних ресурсів по завершенню вирішення користувачем поставлених задач дозволяє збільшити ефективність використання обладнання, адже вивільнений ресурс може бути використаний іншими користувачами шляхом створення тимчасових інформаційних систем для проведення обчислень.

Використана література:

1. *Chen G.* (april 2014) Red Hat Enterprise Virtualization. White paper. IDC [] – Режим доступа : <http://www.redhat.com/en/files/resources/en-rhev-idc-whitepaper.pdf>
2. *Волков Д.* На пороге новой платформы // Computerworld Online – 2014. – № 20. – Режим доступа : <http://www.osp.ru/news/articles/2014/20/13041077/>
3. *Дубова Н.* Академия выбирает облако // Открытые системы. СУБД – 2013. – № 1. – Режим доступа : <http://www.osp.ru/os/2012/01/13012920/>
4. *Коноплев В., Назиров Р.* “Вычислительное облако” как средство эффективной организации вычислительных ресурсов в центре обработки научных данных / В. Коноплев, Р. Назиров // Информационное общество. – 2013. – № 1-2. – С. 17-25.
5. *Kundra V.* Federal cloud computing strategy // The White House. Washington. – February 8, 2011. Режим доступа : <https://www.dhs.gov/sites/default/files/publications/digital-strategy/federal-cloud-computing-strategy.pdf>
6. *Badger L., Grance T., Patt-Corner R., Voas J.* Cloud Computing Synopsis and Recommendations. Recommendations of the National Institute of Standards and Technology. NIST Special Publication 800-146. Available at: <http://csrc.nist.gov/publications/nistpubs/800-146/sp800-146.pdf>
7. *Andrew M. St. Laurent,* Understanding Open Source and Free Software Licensing // O'Reilly Media – 2004. – 224 pages.
8. OpenStack Foundation (12 august 2014) OpenStack Cloud Administrator Guide. Available at: <http://docs.openstack.org/admin-guide-cloud/admin-guide-cloud.pdf>

9. *T. Fifield, D. Fleming, A. Gentle, L. Hochstein, J. Proulx, E. Toews, J. Topjian.* OpenStack Operations Guide // O'Reilly Media – 2014. – 219 pages.
10. OpenStack Foundation (8 august 2014) OpenStack High Availability Guide. Available at: <http://docs.openstack.org/admin-guide-cloud/admin-guide-cloud.pdf>
11. *Jansen W., Grance T.* Guidelines on Security and Privacy in Public Cloud Computing, NIST Special Publication 800-144. Available at: <http://csrc.nist.gov/publications/nistpubs/800-144/SP800-144.pdf>
12. Joint Task Force Transformation Initiative (April, 2013) Recommended Security Controls for Federal Information Systems and Organizations, NIST Special Publication 800-53, Revision 4. Available at: <http://dx.doi.org/10.6028/NIST.SP.800-53r4>

References:

1. *Chen G.* (april 2014) Red Hat Enterprise Virtualization. White paper. IDC. – Режим доступа: <http://www.redhat.com/en/files/resources/en-rhev-idc-whitepaper.pdf>
2. *Volkov D.* Na poroge novoy platformy // Computerworld Online. – 2014. – № 20. – Режим доступа : <http://www.osp.ru/news/articles/2014/20/13041077/>
3. *Dubova N.* Akademiya vybiraet oblako // Otkrytye sistemy. SUBD – 2013. – № 1. – Режим доступа : <http://www.osp.ru/os/2012/01/13012920/>
4. *Konoplev V., Nazirov R.* “Vychislitelnoe oblako” kak sredstvo effektivnoy organizatsii vychislitelnykh resursov v tsentre obrabotki nauchnykh dannykh // Informatsionnoe obshchestvo. – 2013. – № 1-2. – S. 17-25.
5. *Kundra V.* Federal cloud computing strategy // The White House. Washington. – February 8, 2011. – Режим доступа : <https://www.dhs.gov/sites/default/files/publications/digital-strategy/federal-cloud-computing-strategy.pdf>
6. *Badger L., Grance T., Patt-Corner R., Voas J.* Cloud Computing Synopsis and Recommendations. Recommendations of the National Institute of Standards and Technology. NIST Special Publication 800-146. Available at: <http://csrc.nist.gov/publications/nistpubs/800-146/sp800-146.pdf>
7. *Andrew M. St. Laurent,* Understanding Open Source and Free Software Licensing // O'Reilly Media – 2004. – 224 pages.
8. OpenStack Foundation (12 august 2014) OpenStack Cloud Administrator Guide. Available at: <http://docs.openstack.org/admin-guide-cloud/admin-guide-cloud.pdf>
9. *T. Fifield, D. Fleming, A. Gentle, L. Hochstein, J. Proulx, E. Toews, J. Topjian.* OpenStack Operations Guide // O'Reilly Media – 2014. – 219 pages.
10. OpenStack Foundation (8 august 2014) OpenStack High Availability Guide. Available at: <http://docs.openstack.org/admin-guide-cloud/admin-guide-cloud.pdf>
11. *Jansen W., Grance T.* Guidelines on Security and Privacy in Public Cloud Computing, NIST Special Publication 800-144. Available at: <http://csrc.nist.gov/publications/nistpubs/800-144/SP800-144.pdf>
12. Joint Task Force Transformation Initiative (April, 2013) Recommended Security Controls for Federal Information Systems and Organizations, NIST Special Publication 800-53, Revision 4. Available at: <http://dx.doi.org/10.6028/NIST.SP.800-53r4>

Лозинский А. П. Обзор функциональных возможностей программного обеспечения облачной платформы Open Stack Icehouse.

В статье отображена информация о функциональных возможностях облачной вычислительной платформы OpenStack Icehouse с открытым программным кодом. Приведен перечень программных служб, которые формируют архитектуру и функциональные возможности платформы OpenStack Icehouse. В частности отображаются функциональные возможности оркестрации, механизмы обеспечения высокой доступности и надежности облачной платформы, методы обеспечения безопасности облачных вычислений.

Ключевые слова: *распределенная вычислительная система, облачная вычислительная платформа, оркестрация, динамическая архитектура, виртуальные машины, высокая доступность, надежность, безопасность облачных вычислений.*

LOZINSKYI A. P. Overview of the functionality of the cloud software platform Open Stack Icehouse.

The article highlights the information about the functionality of the project for the open-source cloud computing platform OpenStack Icehouse. The article shows the list of the application services that form the architecture and functionality of the OpenStack Icehouse platform. In particular the article highlights orchestration functionality and mechanisms for ensuring high availability and reliability of cloud platforms, methods of cloud computing security.

Keywords: distributed computing system, cloud computing platform, orchestration, dynamic architecture, virtual machines, high availability, reliability, cloud computing security.

УДК 37.091.12.011.3-051:62/65:378:373.5

Ліпчанська І. М.

ВЗАЄМОДІЯ ВИЩОГО І ЗАГАЛЬНООСВІТНЬОГО НАВЧАЛЬНИХ ЗАКЛАДІВ ЯК УМОВА УДОСКОНАЛЕННЯ ПРОФЕСІЙНОЇ ПІДГОТОВКИ МАЙБУТНІХ УЧИТЕЛІВ ТЕХНОЛОГІЙ

У статті розглянута взаємодія вищого педагогічного і загальноосвітнього навчальних закладів з метою удосконалення професійної підготовки майбутніх вчителів технологій.

Ключові слова: взаємодія вищого педагогічного навчального закладу та загальноосвітнього навчального закладу, форми та напрямки взаємодії, вчитель технологій, професійна підготовка.

Соціально-економічні і політичні зміни в Україні, її поступова інтеграція до європейського і світового освітнього простору висувають перед вітчизняною системою освіти нові вимоги, серед яких особливо великого значення набуває удосконалення підготовки майбутніх учителів. У проєкті Концепції розвитку освіти на період 2015–2020 років зазначено, що одним з пріоритетних напрямків реформування освіти є реформування системи підготовки й перепідготовки педагогічних та управлінських кадрів в освітньому секторі [5]. Отже підготовка педагогічних кадрів та їхнє професійне вдосконалення – основні умови модернізації освіти, які вимагають термінових змін. В зв'язку з цим особливої уваги заслуговує проблема взаємодії вищого педагогічного і загальноосвітнього навчальних закладів як необхідна умова підвищення якості підготовки майбутніх педагогів, а також безперервного підвищення кваліфікації педагогічних працівників [3].

Аналіз наукових праць показав, що взаємодія навчальних закладів різних рівнів цікавить багатьох науковців. Так, дослідженнями проблем наступності загальноосвітньої та профільної школи займалися В. Алфімов, Р. Ільясов, В. Кожевников, А. Литвин, М. Дідовик, Н. Муранова, Л. Тютюн,