УДК 004.056
DOI: 10.31673/2409-7292.2022.025259

Яновський Д.В., Тищенко В.С.

ЗАХИСТ ХМАРНОЇ ІНФРАСТРУКТУРИ ВІД КІБЕРАТАК

У статті проведено аналіз сучасних хмарних систем. Досліджено проблеми, конфлікти і можливі обмеження хмарного програмного забезпечення, а також хмарних сервісів. Розроблено рекомендації щодо забезпечення захисту.

Ключові слова: хмарні технології, мережа, хмарні обчислення, сервери, маршрут, хмарні сервіси, модель розгортання хмар, цифровий контент, інформація.

Вступ

Сьогодні користувачі будь-якої із існуючих операційних систем постійно використовують хмарні технології. За останні роки ця тема стала однією із найпопулярніших в IT-сфері, про неї написано чимало статей, проведено ще більшу кількість конференцій, а число існуючих рішень, які використовуються у повсякденному житті багатьма користувачами, майже не можливо перерахувати.

Хмарні технології допомагають досягти як значної економії, так і підвищення гнучкості. Однак при впровадженні цих технологій на підприємствах не слід залишати без уваги традиційні процеси — їх необхідно оптимізувати, враховуючи нові умови. Через відсутність фізичного доступу до серверів в публічних хмарних середовищах питання безпеки стають ще більш важливими, адже в надзвичайній ситуації не буде можливості натиснути кнопку екстреного відключення.

Мета статті – підвищення ефективності безпеки даних в хмарному середовищі.

Існуючі загрози хмарної безпеки

Як відомо, кількість хмарних міграцій з кожним роком зростає, а питання безпеки як і раніше залишається серйозною темою. Першим кроком до мінімізації ризиків в хмарі є своєчасне визначення ключових загроз безпеки. На конференції RSA, CSA (Cloud Security Alliance) представила список загроз хмарної безпеки, з якими стикаються організації. CSA - некомерційна організація, лідер в області стандартів, рекомендацій та ініціатив, спрямованих на підвищення безпеки і захищеності використання хмарних обчислень.

Існують такі загрози хмарної безпеки:

Витік даних. Хмара піддається тим же загрозам, що і традиційні інфраструктури. Через велику кількість даних, які сьогодні часто переносяться в хмари, площадки хмарних хостинг-провайдерів стають привабливою метою для зловмисників. При цьому серйозність потенційних загроз безпосередньо залежить від важливості і значимості даних, що зберігаються.

Компрометація облікових записів і обхід автентифікації. Витік даних найчастіше є результатом недбалого ставлення до механізмів організації перевірки автентичності, коли використовуються слабкі паролі, а управління ключами шифрування і сертифікатами відбувається неналежним чином. Крім того, організації стикаються з проблемами управління правами та дозволами, коли кінцевим користувачам призначаються значно більші повноваження, ніж в дійсності необхідно.

Злом інтерфейсів і API. На сьогоднішній день хмарні сервіси і додатки немислимі без зручного для користувача інтерфейсу. Від того, наскільки добре опрацьовані механізми контролю доступу, шифрування в API, залежить безпека і доступність хмарних сервісів. При взаємодії з третьою стороною, що використовує власні інтерфейси API, ризики значно зростають. Тому що потрібно надавати додаткову інформацію, аж до логіна і пароля користувача. Слабкі з точки зору безпеки інтерфейси стають важливим місцем в питаннях доступності, конфіденційності, цілісності та безпеки.

Вразливість використовуваних систем. Уразливість використовуваних систем - проблема, яка трапляється в мультіарендних хмарних середовищах. На щастя, вона мінімізується шляхом правильно підібраних методів управління IT. Кращі практики

© *Яновський Д.В.,Тищенко В.С., 2022*

включають в себе регулярне сканування на виявлення вразливостей, застосування останніх патчів і швидку реакцію на повідомлення про загрози безпеці.

Викрадення облікових записів. Фішинг, шахрайство, експлойти зустрічаються і в хмарному оточенні. Сюди додаються загрози у вигляді спроб маніпулювати транзакціями і змінювати дані. Хмарні площадки розглядаються зловмисниками як поле для здійснення атак. І навіть дотримання стратегії «захисту в глибину» може виявитися недостатнім.

Інсайдери-зловмисники. Інсайдерська загроза може виходити від нинішніх або колишніх співробітників, системних адміністраторів, підрядників або партнерів по бізнесу. Інсайдери-зловмисники переслідують різні цілі, починаючи від крадіжки даних до бажання просто помститися. У випадку з хмарою мета може полягати в повному або частковому руйнуванні інфраструктури, отриманні доступу до даних та інше. Системи, що прямо залежать від засобів безпеки хмарного постачальника, - великий ризик.

Цільові кібератаки. Розвинена стійка загроза, або цільова кібератака, - в наш час не рідкість. Володіючи достатніми знаннями і набором відповідних інструментів, можна домогтися результату. Зловмисника, який поставив за мету встановити і закріпити власну присутність в цільовій інфраструктурі, не так легко виявити. Для мінімізації ризиків та профілактики подібних загроз постачальники хмарних послуг використовують просунуті засоби безпеки. Але крім сучасних рішень, потрібне розуміння сутності і природи такого виду атак.

Перманентна втрата даних. Оскільки хмари стали досить зрілими, випадки з втратою даних без можливості відновлення через постачальника послуг вкрай рідкісні. При цьому зловмисники, знаючи про наслідки перманентного видалення даних, мають на меті вчинення подібних деструктивних дій. Хмарні хостинг-провайдери для дотримання заходів безпеки рекомендують відокремлювати призначені для користувача дані від даних додатків, зберігаючи їх в різних локаціях. Не варто забувати і про ефективні методи резервного копіювання. Щоденний бекап і зберігання резервних копій на зовнішніх альтернативних захищених площадках особливо важливі для хмарних середовищ.

Недостатня поінформованість. Організації, які переходять в хмару без розуміння хмарних можливостей, стикаються з ризиками. Якщо, наприклад, команда розробників з боку клієнта недостатньо знайома з особливостями хмарних технологій і принципами розгортання хмарних додатків, виникають операційні та архітектурні проблеми.

Зловживання хмарними сервісами. Хмари можуть використовуватися легітимними і нелегітимними організаціями. Мета останніх - використовувати хмарні ресурси для здійснення зловмисних дій: запуску DDoS-атак, відправки спаму, поширення шкідливого контенту і т. Д. Постачальникам послуг вкрай важливо вміти розпізнавати таких учасників, для чого рекомендується детально вивчати трафік і використовувати інструменти моніторингу хмарних середовищ.

Ddos-атаки. Незважаючи на те що DoS-атаки мають давню історію, розвиток хмарних технологій зробило їх більш поширеними. В результаті DoS-атак може сильно сповільнитися або зовсім припинитися робота значущих для бізнесу компанії сервісів. Відомо, що DoS-атаки витрачають велику кількість обчислювальних потужностей, за використання яких буде платити клієнт. Незважаючи на те що принципи DoS-атак, на перший погляд, прості, необхідно розуміти їх особливості на прикладному рівні: вони націлені на уразливості веб-серверів і баз даних. Хмарні постачальники, безумовно, краще справляються з DoS-атаками, ніж окремо взяті клієнти.

Спільні технології, загальні ризики. Уразливості в використовуваних технологіях - достатня загроза для хмари. Постачальники хмарних послуг надають віртуальну інфраструктуру, хмарні додатки, але якщо на одному з рівнів виникає вразливість, вона впливає на все оточення. [17].

Методи забезпечення безпеки в хмарі

Для забезпечення безпеки хмарних обчислень використовується широкий набір технологій та управлінських механізмів, розгорнутих для захисту даних, додатків і пов'язаної

© *Яновський Д.В.,Тищенко В.С., 2022*

з ними інфраструктури хмарних обчислень. Деякі передові алгоритми шифрування, які були застосовані в хмарних обчислень збільшують захист приватного життя. Організація адекватного контролю доступу, використання механізмів багатофакторної автентифікації, включаючи одноразові паролі, токени, смарт-карти, USB-ключі, а також користування інструментами захисту і раннього виявлення загроз – все це збільшує рівень безпеки хмарних середовищ.

Шифрування. Шифрування само по собі не перешкоджає втручанню, але заперечує прозорий контент для потенційного перехоплювача. У схемі шифрування передбачувана інформація або повідомлення, іменована відкритим текстом, шифруються за допомогою алгоритму шифрування - шифрувального тексту, що генерує шифр, який може бути прочитаний, тільки після розшифрування. З технічних причин схема шифрування зазвичай використовує псевдовипадковий ключ шифрування, що генерується алгоритмом. В принципі, можна розшифрувати повідомлення, не маючи ключа, але для добре продуманої схеми шифрування необхідні значні обчислювальні ресурси та навички. Уповноважений одержувач може легко розшифрувати повідомлення за допомогою ключа, наданого одержувачем, а не завдяки неавторизованим користувачем.

Існують два методи шифрування - симетричне та асиметричне. В основному, симетричні алгоритми шифрування вимагають менше обчислень, ніж асиметричні. На практиці, це означає, що якісні асиметричні алгоритми в сотні або в тисячі разів повільніші за якісні симетричні алгоритми. Недоліком симетричних алгоритмів є необхідність мати секретний ключ з обох боків передачі інформації. Так як ключі є предметом можливого перехоплення, їх необхідно часто змінювати та передавати по безпечних каналах передачі інформації під час розповсюдження [18].

При застосуванні із асиметричними алгоритмами шифрування для передачі ключів, майже завжди використовуються генератори криптографічно стійких псевдовипадкових чисел для генерування симетричних ключів сеансу. Однак, брак достатнього рівня випадковості в цих генераторах, або в їх початкових векторах, в минулому часто призводив до втрати конфіденційності при передачі даних. Ретельний підхід до впровадження криптосистеми та генерація випадкових чисел із використанням високоякісних джерел випадкових чисел є дуже важливими для збереження конфіденційності даних, що передаються.

Асиметричні криптосистеми — ефективні системи криптографічного захисту даних, які також називають криптосистемами з відкритим ключем. В таких системах для зашифровування даних використовують один ключ, а для розшифровування — інший (звідси і назва — асиметричні). Перший ключ є відкритим і може бути опублікованим для використання усіма користувачами системи, які шифрують дані. Розшифровування даних за допомогою відкритого ключа неможливе. Для розшифровування даних отримувач зашифрованої інформації використовує другий ключ, який є секретним (закритим). Зрозуміло, що ключ розшифровування не може бути визначеним з ключа зашифровування.

Рис. 1. Принцип роботи асиметричної криптосистеми

© *Яновський Д.В.,Тищенко В.С., 2022*

Проблемою симетричного шифрування є необхідність передачі ключа, для розшифрування інформації, таким чином ключ може бути перехоплений кимось іншим. Будь хто, знаючи секретний ключ, може розшифрувати інформацію. Тоді як в асиметричному шифруванні є два пов'язаних ключа — пара ключів. Відкритий ключ — публічний, до нього повинні мати доступ всі ті, хто матиме потребу зашифрувати інформацію. Тоді як закритий ключ — приватний ключ, повинен бути доступним лише тому хто має право розшифрувати інформацію, за своїм розміром він значно більший від секретного ключа симетричного шифрування. Будь-яку інформацію, зашифровану за допомогою відкритого ключа можна розшифрувати лише застосовуючи той самий алгоритм, але з використанням відповідного приватного ключа. Також всю інформацію, зашифровану за допомогою приватного ключа, можна розшифрувати лише за допомогою відповідного відкритого ключа. Це означає, що немає необхідності хвилюватись за передачу ключа, відкритий ключ повинен бути публічним. Але асиметричне шифрування є значно повільнішим від симетричного. Також потребує значно більше обчислювальної потужності як для шифрування, так і для розшифрування інформації.

Токен автентифікації. Токени (також апаратний токен, USB-ключ, криптографічний токен) призначені для електронного посвідчення особи (наприклад, клієнта, який одержує доступ до банківського рахунку), при цьому вони можуть використовуватися замість або разом з паролем. У певному сенсі токен — це електронний ключ для доступу до чого-небудь. Зазвичай, це компактний фізичний пристрій, призначений для забезпечення інформаційної безпеки користувача, використовується для ідентифікації його власника, безпечного віддаленого доступу до інформаційних ресурсів, а також для спрощення автентифікації.

Найпростіша вразливість з будь-яким токеном — це його втрата або крадіжка. Імовірність випадку компрометації може бути зменшена за допомогою особистої безпеки, наприклад: замки, електронна прив'язь, сигналізація. Вкрадені токени — марні для злодія, якщо використана технологія двофакторної автентифікації. Зазвичай для перевірки автентичності потрібно вводити персональний ідентифікаційний номер (PIN) разом з інформацією на токені.

Багатофакторна автентифікація. Багатофакторна автентифікація (БФА, англ. multi-factor authentication, MFA) — розширена автентифікація, метод контролю доступу до комп'ютера, в якому користувачеві для отримання доступу до інформації необхідно пред'явити більше одного «доказу механізму автентифікації». Двофакторна автентифікація (ДФА, англ. two-factor authentication, також відома як двоетапна верифікація), є типом багатофакторної автентифікації. ДФА — представляє собою технологію, що забезпечує ідентифікацію користувачів за допомогою комбінації двох різних компонентів.

Хорошим прикладом двофакторної автентифікації є авторизація Google і Microsoft. Коли користувач заходить з нового пристрою, крім автентифікації по імені та паролю, його просять ввести шестизначний (Google) або восьмизначний (Microsoft) код підтвердження. Отримати його можна за допомогою SMS, або голосового дзвінка на мобільний телефон, він може бути взятий із заздалегідь складеного реєстру разових кодів або користувач може використати додаток-автентифікатор, генеруючий новий одноразовий пароль за короткі проміжки часу. Вибрати один з методів можна в налаштуваннях Google або Microsoft-акаунта.

Зараз майже всі великі сервіси, такі як Microsoft, Google, Yandex, Dropbox, Facebook, вже надають можливість використовувати двофакторну автентифікацію. Причому для всіх з них можна використовувати єдиний додаток автентифікатор, що відповідає певним стандартам, такі як Google Authenticator, Microsoft Authentificator, Authy або FreeOTP.

Наприклад, щоб отримати доступ до хмарного сховища iCloud від Apple, користувачу необхідно підтвердити автентифікацію завдяки мобільному пристрою. Переглянути збережені у хмарі файли можна завдяки веб-версії сервісу, відкривши її у будь-якому браузері.

© *Яновський Д.В.,Тищенко В.С., 2022*

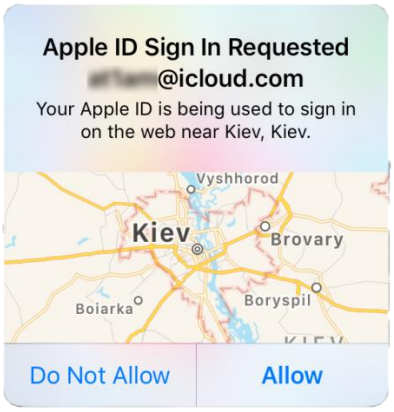

Рис. 3. Запит дозволу для авторизації

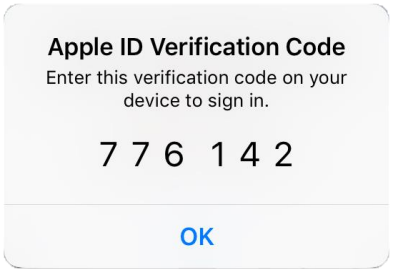

Рис. 4. Згенерований шестизначний код

Якщо користувач дає дозвіл – генерується шестизначний код, який необхідно ввести для успішної авторизації. Завдяки двохфакторній автентивфікації можна значно підвищити рівень захищенності даних у хмарі. Таким чином зловмисник не зможе отримати доступ до хмарного сховища, навіть знаючи електронну адресу та пароль.

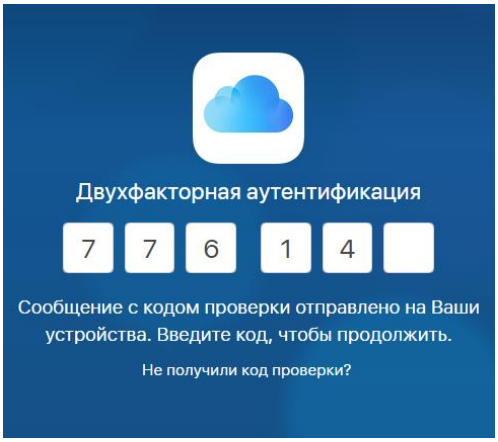

Рис. 5. Процес авторизації з використанням двохфакторної автентифікації

Переваги використання двофакторної автентифікації через мобільний пристрій очевидні: не потрібні додаткові токени, тому що мобільний пристрій завжди під рукою, а також код підтвердження постійно змінюється, що значно безпечніше, ніж однофакторний логін-пароль.

З недоліків двофакторної автентифікації через мобільний пристрій потрібно відмітити:

– Мобільний телефон повинен ловити мережу, коли відбувається автентифікація, інакше повідомлення з паролем просто не дійде.

– Ви ділитеся з кимось вашим мобільним телефоном, що впливає на ваше особисте життя і може бути в майбутньому на нього буде приходити спам.

– Текстові повідомлення (SMS), які, потрапляючи на ваш мобільний телефон, можуть бути перехоплені.

– Текстові повідомлення приходять з деякою затримкою, так як деякий час йде на перевірку.

– Сучасні смартфони використовуються як для одержання пошти, так і для отримання SMS. Як правило електронна пошта на мобільному телефоні завжди включена. Таким чином, усі акаунти, для яких пошта є ключем, можуть бути зламані (перший фактор). Мобільний пристрій (другий фактор). Висновок: смартфон змішує два чинника в один.

© *Яновський Д.В.,Тищенко В.С., 2022*

Однак не всі зазначені недоліки відносяться до розглянутого сервісу хмарного сховища iCloud. Як було продемонстровано у прикладі, шестизначний код відправляється на мобільний пристрій не завдяки електронній пошті або SMS, а напряму у вигляді сповіщення, що унеможливлює перехоплення такого повідомлення.

Резервне копіювання. Резервне копіювання або бекап (англ. backup) — процес створення копії даних з носія, призначений для відновлення цих даних у разі їх пошкодження або видалення. Створення резервної копії даних надає можливість виконати відновлення інформації при втраті оригіналу, з якого було створено резервну копію. При цьому під втратою треба розуміти настання події, що призвела до зміни даних, після чого вони втратили цінність або були видалені з носія. Приклад: умисне завдання шкоди через видалення важливої для підприємства інформації. Об'єкти резервного копіювання — це дані або сукупність даних, з яких можна створити резервну копію. Приклади об'єктів: файли або теки, дані прикладних програм, дані операційної системи чи сама ОС (наприклад Windows System State або AIX System Backup), образи віртуальних машин та дисків віртуальних машин, файлові системи тощо.

Для визначення вимог для швидкості відновлення та періодичності створення резервних копій, використовують наступні визначення:

Recovery Point Objective (RPO) - Цільова точка відновлення. Визначає періодичність створення резервних копій. У разі пошкодження або видалення даних і потребу відновити "найсвіжішу" копію, це можливо зробити лише на останню RPO. В залежності від технологій, що були використані під час створення резервних копій, відновлення можливе також у період між різними RPO, така технологія називається Point In Time Restore та властива СУБД.

Recovery Time Objective (RTO) - Цільовий час відновлення. Визначає час, який повинен пройти з початку і до кінця відновлення даних з резервної копії. Час з моменту втрати даних до початку роботи спеціаліста по відновленню, а також час з моменту відновлення з резервної копії до початку використання даних користувачами, може не включатися до RTO.

Для можливості відновлення під час катастрофи (відмова сайту або парна відмова зарезервованих компонентів) використовують також такі визначення:

Disaster Recovery Point Objective (DRPO) - Цільова точка відновлення під час катастрофи. RPO у разі настання катастрофи.

Disaster Recovery Time Objective (DRTO) - Цільовий час відновлення під час катастрофи. RTO у разі настання катастрофи.

Існують наступні рівні резервного копіювання:

– Повне резервне копіювання (Full Backup або L0) — повна копія даних. Рівень, який забезпечує створення повної копії об'єкту резервного копіювання. Цей рівень дозволяє забезпечити максимальну відповідність оригіналу даних його копії.

– Диференційне резервне копіювання (Differential Backup або L1) — копіювання змін, що були зроблені після створення останньої повної копії. Створення такої копії потребує більше часу та займає більший об'єм, ніж додаткове копіювання, але дозволяє пришвидшити процес відновлення. Загалом є альтернативою між створенням повної або додаткової копії.

– Додаткове резервне копіювання (Incremental Backup або L2) — копіювання змін, що відбулись із часу повного, диференційного або додаткового копіювання. Загалом на додаткове копіювання затрачається менше часу, бо копіюється менше файлів. Однак процес відновлення даних займає більше часу, оскільки повинні спочатку відновлюватися дані останньої повної копії і після цього — всі резервні копії, від яких залежить додаткова копія.

– Lx (розшифровується як Level X) — Це один із стандартів описання методів резервного копіювання. Оскільки різні джерела можуть по різному тлумачити поняття диференційного та додаткового резервного копіювання, може використовуватись нотація Lx, що для перелічених типів буде означати L1 та L2 відповідно. Рівень Lx завжди залежить від

© *Яновський Д.В.,Тищенко В.С., 2022*

попереднього Lx-1 або Lx-n, якщо він був останнім. Наприклад, у разі наступної послідовності резервних копій L0, L5, L3, L2, L4, процедура відновлення на останню найсвіжішу резервну копію буде відбуватись у такій послідовності: L0, L2, L4.

При своєчасному резервному копіюванні даних у хмарі можна запобігти втраті зберігаємої інформації.

Висновки

Архітектура безпеки хмари є ефективною, тільки якщо правильно реалізовано захист на місці. Ефективна архітектура безпеки хмари визначає проблеми, які виникатимуть з керуванням безпеки. Управління безпеки усуває проблеми пов'язані з контролем безпеки. Ці елементи управління вступають в дію для захисту будь-яких недоліків в системі і зменшення впливів атак.

Рекомендується, що контроль інформаційної безпеки повинен бути вибраний і реалізований відповідно і в пропорції до ризиків, як правило, шляхом оцінки загроз, вразливостей і впливів. Проблеми безпеки хмари можуть бути згруповані різними способами, серед них присутні такі: управління ідентифікацією, фізична безпека, безпека персоналу, доступність, безпека додатків, приватність.

Перелік посилань

1. Haghighat, Mohammad (2015). CloudID: Trustworthy Cloud-based and Cross-Enterprise Biometric Identification. Expert Systems with Applications 42 (21). – P. 7905–7916.
2. Eric Grosse, Mayank Upadhyay, Authentication at Scale. IEEE Security and Privacy, January/February 2013, IEEE Computer and Reliability Societies. – P. 15-22
3. Gnanasundaram, S.; Shrivastava, A., eds. (2012). Information Storage and Management: Storing, Managing, and Protecting Digital Information in Classic, Virtualized, and Cloud Environments. John Wiley and Sons. – 255 p.
4. Pritchard, S. (December 2017). "Cloud-to-cloud backup: What it is and why you need it". Computer Weekly. TechTarget. – Режим доступу: World Wide Web. – URL: https://www.computerweekly.com/feature/Cloud-to-cloud-backup-What-it-is-and-why-you-need-it
5. Wayne Jansen, Timothy Grance Guidelines on Security and Privacy in Public Cloud Computing, NIST Special Publication 800-144, 2011. – Режим доступу: World Wide Web. – URL: http://csrc.nist.gov/publications/nistpubs/800-144/SP800- 144.pdf

Надійшла: 12.06.2022 Рецензент: д.т.н., доцент Ахрамович В.М.

© *Яновський Д.В.,Тищенко В.С., 2022*