Осигуряване на сигурността в облачна система чрез шаблони за проектиране

Мария Армянова

Providing the Cloud Security by Design Patterns

Mariya Armyanova

Abstract

The problem of providing the security of cloud services is becoming more popular as the use and availability of cloud services increases. Besides small and medium-sized businesses, and governments use cloud technology to reduce information costs and increase the availability and scope of offering services. Cloud technologies are expected to meet the increased security requirements. Enhanced requirements makes difficult development of cloud computing sevices and infrastructure. One way to overcome these difficulties is to use design patterns that aid applying of useful security practices. For their application is necessary a research of the variety of design patterns offered by different organizations and scientists. That's why the research object in the paper is design patterns that support improving cloud security. Some security issues and design patterns that aid to solve them are addressed.

Keywords: design pattern, security, cloud computing, cryptographic design patterns, secure cloud interfaces, cloud data protection.

JEL Code: C88

Въведение

Облачните технологии имат възможността да намалят разходите и сложността на обезпечаване на ресурси за реализация на нуждите на потребителите в областта на информационните технологии. Чрез тях информационните системи могат да се реализират без предварителни инвестиции в инфраструктура, софтуерни лицензи и др. Затова тенденцията е за нарастване използването и предлагането на облачни услуги, представени чрез широк спектър от приложения, вариращи от високо изчислително интензивни приложения до такива нуждаещи се от съвсем малко ресурси. Освен малки и средни предприятия и правителствата използват възможностите на облачната технология за намаляване на разходите за информационно осигуряване и увеличаване на достъпността и обхвата на предлаганите от тях услуги. От облачните технологии се очаква да отговорят на завишените изисквания към сигурността на предлаганите услуги.

Облачната технология се реализира посредством множество участници с различни роли, като доставчици на инфраструктура за потребителите в облака, доставчици на услуги или приложения за крайни потребители и потребители на услугите, хоствани на инфраструктурата на облака. Всеки от тях има свои възможности за управление на сигурността и собствени изисквания към нея, като същевременно разчита и на определено ниво на сигурност, предоставено от другите участници. Именно обвързаността на различните участници налага познаване и спазване на множество изисквания и стандарти при реализацията на механизмите за сигурност. Това затруднява разработчиците на облачните приложения и инфраструктура. Начин да се преодолеят трудностите е използването на шаблоните за проектиране, които капсулират полезни практики в областта на сигурността. За да се използват обаче е необходимо да се познава многообразието от шаблони, предлагани от различните организации и учени. Цел на изследването е да се открият често срещани проблеми на осигуряването на сигурността и шаблони, които подпомагат решаването им.

1. Проблеми със сигурността на облачните услуги

Има много проблеми, свързани със сигурността, които следва да са обект на внимание на облачните системи. Те произтичат от предлаганите в Интернет широка гама облачни

услуги в областта на хардуера и софтуера. Все пак като основен проблем при предлагането на облачни услуги може да се определи осигуряване на поверителността и сигурността на данните на клиентите. Често в медийното пространство нашумяват скандали, свързани с нарушаване на поверителността на данните на клиентите, при това на големи корпорации, като Гугъл, Британ еъруейз и др. Понякога подобно нарушение на поверителността е преднамерено, например в случая, разпространен от Блумбърг, с компанията Мастеркард, която е продала данни за клиентите си в Америка на компанията Гугъл с цел по-ефективна реклама. Но в повечето случаи подобно извличане на клиентски данни е резултат от умели хакерски атаки. Причината за допускането на пропуски в сигурността е бързината на разработката на облачната система, подпомагаща навлизането на големите компании на новите дигитални пазари. Водещо при разработването на софтуера са нуждите на компаниите, а изискванията и нуждите на клиентите им се пренебрегват. Резултатът е, че компаниите започват да предлагат широк кръг от услуги в Интернет пространството, преди да са разрешени проблемите за поверителността и сигурността на данните на клиентите. Затова някои клиенти, които са чувствителни по отношение на сигурността на данните си, се въздържат да използват възможностите на облачните системи.

Тъй като в облаците се съхранява големи информационни масиви с разнообразно предназначение, те са непрекъсната цел на атаката на хакерите. Затова сериозен проблем е осигуряване сигурността на данни, съхранявани в облаците. За решаването му възникват различни асоциации и организации с не стопанска цел, които да насърчат използването на полезни практики, за осигуряване на сигурност на облачните услуги. Такива е Cloud Security Alliance (CSA), която подпомага създаването на стратегията за сигурност на облаците на американското правителство. А също и европейската European Union Agency for Network and Information Security (ENISA), която определя рисковите ситуации, свързани със защитата на данните в облаците. Тя има за цел да предотврати, открие и реагира на потенциалните проблеми с мрежовата и информационна сигурност.

Доставчиците на облачни услуги, обикновено оставят отговорността за осигуряване на сигурността и надеждността на данните на клиентите. Така всеки отговаря сам за осигуряване на безопасността на данните си в облачното пространство. За да се предпазят от пробив в сигурността на информацията, на клиентите (особено, ако са организации с критични данни) се препоръчва да използват многостранно потвърждение и криптиране. Използването на криптиране от страна на клиента, обаче изисква от доставчиците наличие на инфраструктура, която да позволи цялостно управление на криптографските ключове.

Проблем може да възникне и при прехвърляне на базите с данни на фирмата в облачното пространство, тъй като се изисква да се обновят и засилят и процедурите, осигуряващи сигурността на данните.

Друг проблем е осигуряване на сигурността на данните в мрежата, тъй като данните се пренасят през нея от организацията до облачните услуги, които ги обработват или съхраняват. За да се избегне изтичането на поверителна информация, преминаването на данните през глобалната мрежа, трябва да е защитено чрез надеждно криптиране на мрежовия трафик.

Осигуряването на сигурна автентикация на потребителите също е проблем. Препоръчително е да се намали излагането на частни данни по време на процедурите по автентикация в облака.

Проблем в облачното пространство е и гарантиране, че няма пробив в процедурите по обработка и съхранение на данните, осигурявани от облачните доставчици.

Част от проблемите по сигурността на облачните системи са общи за всички системи. Те са наследени от използвани технологии като виртуализация и SOA[1]. Приложенията

[1] Service-Oriented Architecture

базирани на архитектурата на микро услуги довеждат до натрупване на голямо количество от едновременно работещи микро услуги, което засилва трудностите при мониторинга и управлението на сигурността им. Използването на архитектурата на микро услугите, влияе и на комуникационния модел на компонентите на софтуера, като го променя от модел "едно към едно" в модел "много към много" и добавя още проблеми по отношение на гарантиране на поверителността и надеждността на данните.

Разгледаните проблеми поставят няколко цели пред сигурността. Основната е обезпечаване на поверителност на данните, като се намали риска за изтичане на информация. Друга цел е удостоверяване на почтеността на доставчиците на услуги, т.е. да се гарантира, че няма неразрешено изменение или унищожаване на информация. Цел е осигуряване на наличност на облачната услуга винаги, когато е необходима. И не на последно място като цел е определено и гарантирането на законността и правилността на процедурите за обработка.

2. Шаблони за сигурност в облачните системи

Полезен опит в решаване на проблемите на сигурността в облачното пространство предлагат шаблоните за проектиране. Те подпомагат въвеждането на принципните постановки на сигурността при разработването на облачната инфраструктура и приложения.

Шаблоните за проектиране могат да се разглеждат, като концепция, която предоставя стандартни решения за архитектурни и концептуални софтуерни проблеми. Те описват решения, които са доказали своята полезност в практиката. Те не са, както завършено софтуерно приложение, което директно да се приложи, така и проектно решение, което може веднага да се трансформира в код. Те са по-скоро правила, указания, които формират определена съставна част от структура на проекто-програмното решение. Използват се за разработката на приложения, продукти и софтуерни решения. Въпреки, че често съдържат програмен код, който да скицира заложената в тях идея, той не се прилага директно. Алгоритмите също не се приемат за шаблони за проектиране, защото решават по-скоро проблеми на реализацията на дадено решение, а не проблеми, свързани с проектирането на проекто-програмен проблем. Според основоположникът на шаблоните Ерик Гама (Gamma, 1993, с. 406-431) шаблонът предизвиква разработчиците да мислят по проблема, после да решат и действат ясно и преднамерено. Шаблоните описват популярно решение на даден софтуерен проблем в обобщен вид, за да има по-голямо приложение. Разработчикът адаптира и прилага шаблона според спецификите на решавания проблем.

В литературата е представен широк набор шаблони, включително и такива, осигуряващи сигурността на софтуерните системи. Тези шаблоните обединяват познанията за сигурността и системната структура, като дават възможност за развитие и усъвършенстване на софтуера. Позволяват да се интегрира политиката на сигурност с проектирането и разработването на софтуера.

Шаблоните за сигурност имат за цел да предотвратят случайното вмъкване на уязвимости в кода и да смекчат последствията от тези уязвимости. За разлика от класическите шаблони за проектиране на GoF[2] (Гама, 2004), шаблоните за сигурност разглеждат проблемите на сигурността от разнообразните нива. Нивата варират от архитектурни шаблони, включващи проектиране на ниво цялостна система, до шаблони на ниво на изпълнение, отнасящи се до реализацията на отделни функции или методи в системата.

Йодер и Баркалоу (Yoder & Barcalow, 1998) за първи път описват шаблоните за сигурност, представящи различни аспекти на сигурността. Преди тях Фернандес (Fernandez, 1993) и колегите му представят обектно-ориентирани модели на системи за сигурност без да ги определят, като шаблони, а по-късно описват шаблони за криптография и за контрол на

[2] Gang of Four, голямата четворка – Гама, Хелм, Джонсън и Влисидес, които са основоположници на шаблоните

достъпа (Fernandez, 2005). Екипът на Дохърти (Dougherty, 2009) предлага шаблони за сигурност, като специализира и комбинира вече описани шаблони така, че те да решават проблеми в областта на сигурността.

В съвременните условия вече има представени цели колекции от шаблони за сигурност и на облачните системи. През март 2013 г. Хафиз (Hafiz, 2013) създаде каталог на публикуваните шаблони за сигурност. Шумахер (Schumacher, 2006) предлага шаблони за сигурност, за уеб приложенията, които са приложими и в облачна среда. Голяма част от шаблоните за сигурност могат да се използват и в областта на облачните изчисления. Дара (Dara, 2014) описва два шаблона за симетрично и асиметрично кодиране на достъпа до доставчика на облачните услуги. Шаблоните на Романовски (Romanovsky, 2001) пък са насочени към проблемите на на сигурността от високо ниво, като например как да се подобри сигурността при комуникацията с ненадеждни системи на трети страни и важността на многопластовата защита. Фернандес (Fernandez, Yoshioka 2014) представя два шаблона за изграждане на защитна стена с цел контрол на мрежовия достъп до облачните ресурси. Те подобряват и разширяват съществуващите до този момент шаблони за сигурност (Schumacher, 2006). Шаблоните за сигурност на Хамнер (Hanmer, 2014) подобряват шаблоните му (Hanmer, 2007) за устойчивия на сривове софтуер.

Някои от шаблоните, създадени за да реализират функционалността в облаците, са предназначени за осигуряване на сигурността на приложенията, работещи в облачна среда. Такива шаблони са предложени във всички големи екипи от учени, създаващи шаблони от областта на облачните изчисления. Такива са разработените от PLoP[3], AWS, Microsoft Azure, Google Cloud, групата, предлагаща шаблони на cloudpatterns.org, архитектурните шаблони на Уайлър, шаблоните на Филинг и др.

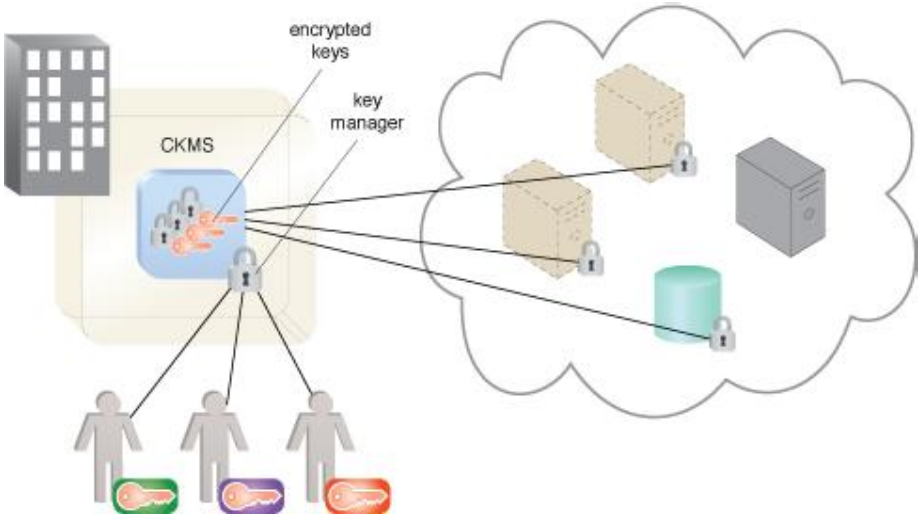
Група от учени от PLoP разработва редица успешно приложими шаблони за реализация на мрежовите ресурси и сигурност. Към тях принадлежат шаблоните за базирана на съобщения интеграция на приложения (Hohpe, 2004), устойчиви на сривове системи (Hanmer, 2007; Hanmer, 2013), или системи с разпределен контрол (Eloranta, 2014).

3. Популярни шаблони за сигурност, използвани в облачните системи

За илюстрация на възможностите на облачните шаблони да решават представените разгледаните проблеми в областта на сигурността е необходимо да се разгледат техни популярни представители и концепциите, заложени в тях. За целта са подбрани шаблони, получени в резултат на сътрудничеството на изявени учени от областта.

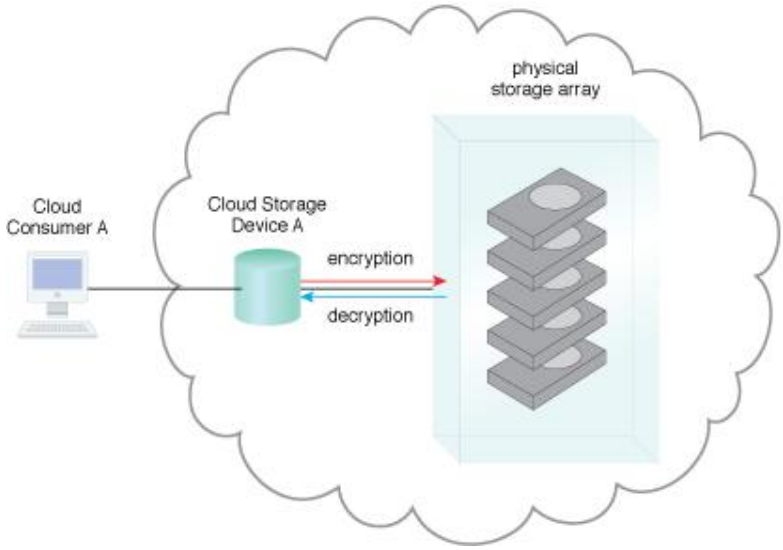
Примери за шаблони, които решават и посочените в т. 1 проблеми са представени от общността cloudpatterns.org (Cloudpatterns.org; Erl, 2015), която обединява множество специалисти от областта на облачните технологии. Техните шаблони не са обвързани с определен доставчик на облачни услуги или технология, затова и описанието им не съдържа конкретно приложение за специфична технология. Криптирането е основополагащо за сигурността на облака, но управлението на криптиращите ключове е едно от най-трудните предизвикателства за реализация в облаците. Неадекватното управление на ключове за криптиране води до редица административни проблеми и проблеми с поверителността на данните, които се съхраняват в облака или сигурността на данните при преноса им. Шаблон, които решава тези проблеми е шаблонът Управление на ключовете в облак (Cloud Key Management). Той предполага използване на система за управление на ключовете в облак, която се реализира чрез физическо или виртуално устройство, свързано към мрежа. Този модел може да се комбинира с други шаблони, като Система за управление на криптографски ключове (Cryptographic Key Management System), шаблони за модули за хардуерна защита (Hardware Security Module patterns), за да се повиши нивото на защита. Представен е на фиг. 1.

[3] Pattern Languages of Programs



Фигура 1. Шаблон Система за управление на криптографски ключове.

Друг шаблон е Криптиране на данните на Rest[4] (Encryption of Data at Rest). Той позволява реализация на защита на данните, съхранявани в облачната среда чрез защита на достъпа до физическите устройства, съхраняващи данни в облака. Той предлага използване на механизъм за криптиране, поддържан от устройствата за данни, който автоматично да кодира данни при съхранението им на дисковете, и да ги декодира при извличането им. Това е начин да се предпазят данните от неправомерен достъп (фиг. 2).

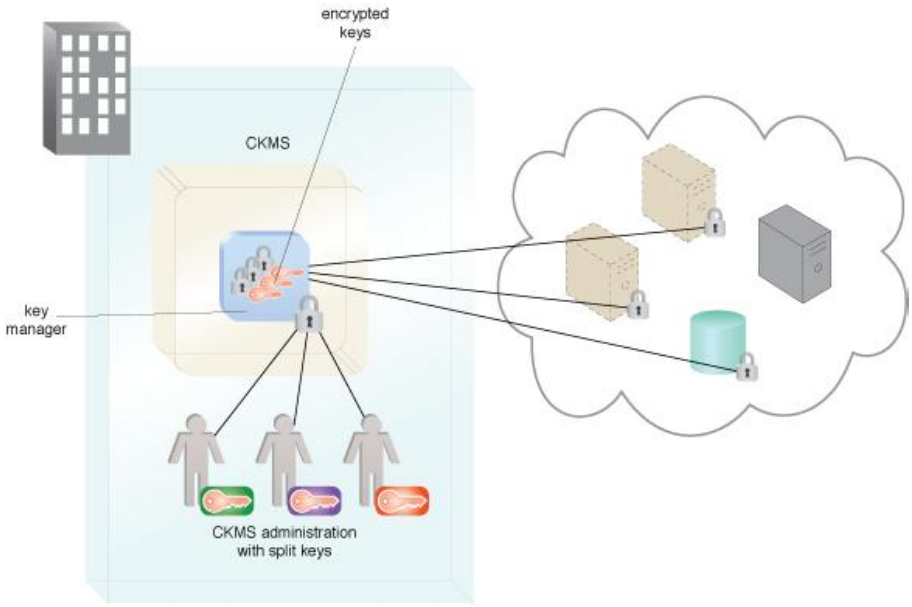


Фигура 2. Шаблон Криптиране на данните на Rest.

Компонентите на облачната система, които осигуряват различни услуги, използват ключове, за да гарантират сигурността на достъпа до услугите. Проблемът е управлението на ключовете в съответствие с регулаторните политики. Шаблонът Управление на ключовете за кодиране(Cryptographic Key Management) предполага въвеждане на системата за управление на криптографските ключове, състояща се от правила, процедури, компоненти и устройства

4 Representational State Transfer

за защита, управление и предоставяне на ключове за кодиране във вид на метаданни. Системата се състои от всички устройства или подсистеми, които имат достъп до некриптиран ключ или метаданни. Криптираните ключове и техните криптографски защитени метаданни могат да се обработват от компютри, да се предават чрез комуникационни системи и да се съхраняват и предават през Интернет и устройства, които не са част системата за управление (фиг. 3).

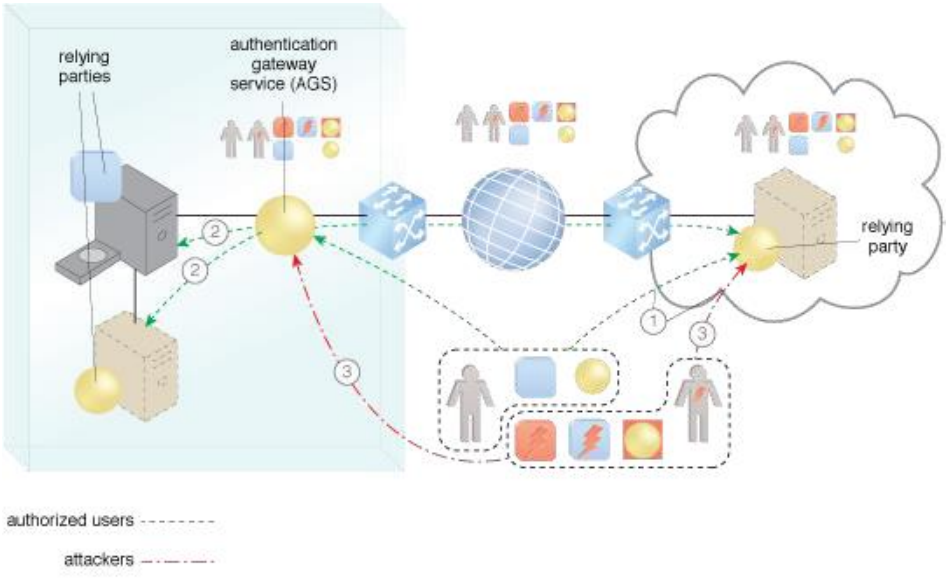


Фигура 3. Шаблон Криптиране на данните на Rest.

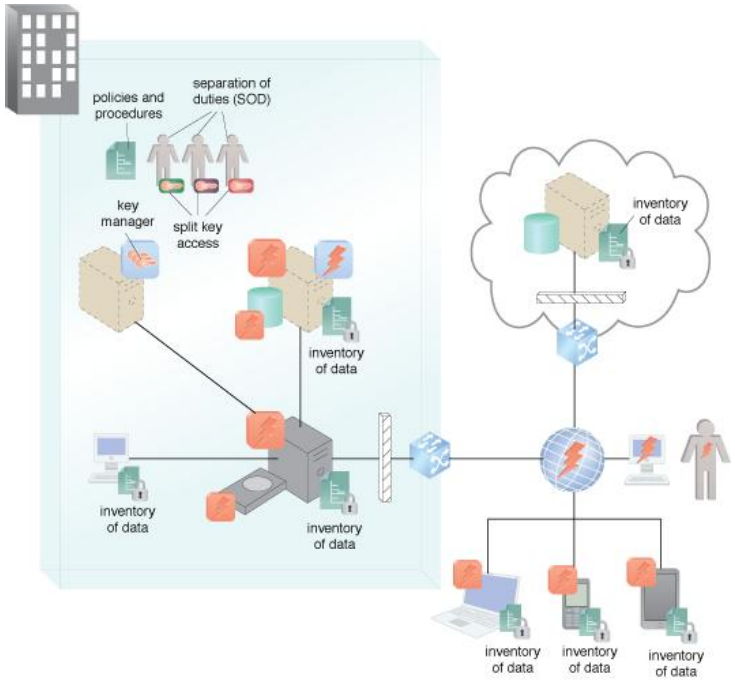
Приложно-програмните интерфейси (API) трябва да предоставят гъвкави интерфейси за сигурност. Шаблон Защита на облачните и API интерфейси (Secure Cloud Interfaces and APIs) се базира на създаване на система за идентификация и управление на достъпа до облака. Целта е да се разграничават легитимните потребители от нарушителите. Предполага се достъп до системата само през точки за идентифициране на достъпа (authentication gateway service AGS), в които следва да се вложат механизми за разпознаване на потребителите. Те също могат да се реализират на база концепциите на други шаблони. Предоставянето на малко и защитени точки за проникване увеличава надеждността на процедурите по автентикация на потребителите (фиг. 4).

Незащитените данни са уязвими на голямо разнообразие от нарушения, които могат да имат значителни последствия за сигурността на облачната архитектура или дейността на клиента на облачната услуга. Решение на проблема е създаване на система, която осигурява кодиране на важните данни, така че дори и данните да станат достояние на лица без право на достъп, те да не могат да ги разкодират без цялостната система и така данните да не са разбираеми. Шаблонът Защита на данните в облака (Cloud Data Breach Protection) дава такава възможност, като обединява криптирането, управлението и политиките за сигурност (фиг. 5).

Още шаблони за сигурност са предложени от Microsoft (microsoft.com) за разработка на системи, поддържани от Windows Azure или друга облачна платформа (Homer, 2014). Те са предназначени за разпределените системи, облачните системи и по-специално за облачната платформа Microsoft Azure. Въпреки, че шаблоните се фокусират върху Azure, авторите твърдят, че представените концепции са общо приложими. Основната цел на шаблоните обаче, е да се подпомогнат разработчиците, създаващи облачна система в съответствие с облачните технологии на Microsoft.



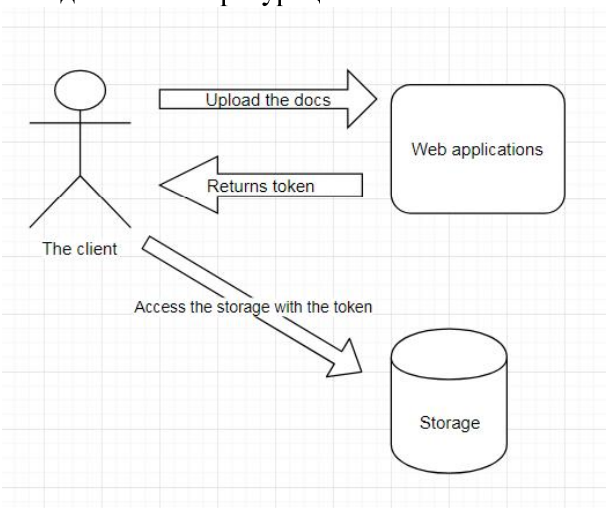
Фигура 4. Шаблон Защита на облачните и приложно-програмните интерфейси.



Фигура 5. Шаблон Защита на данните в облака.

Такъв е шаблонът Ключ за сейф (Vault key), решава проблем за сигурността при качване и изтегляне на данни, които клиентът съхранява в облака. Решението,описвано от шаблона се състои в предоставяне на клиента на временен ключ, генериран по време на изпълнение, а клиентът да има достъп до хранилището на данните си с помощта на тези ключове (tokens) (фиг. 6). Всеки потребител получава достъп само до тези данни от хранилището, защитени от специфичния ключ. Така се поддръжка сигурността при

едновременното използване на една услуга от много клиенти, при която всеки потребител може да вижда само своите данни и конфигурации.



Фигура 6. Шаблон Ключ за сейф.

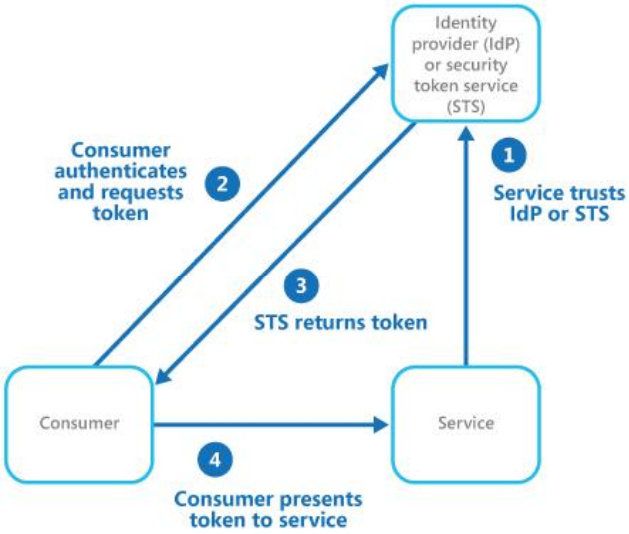
Шаблонът Федеративна идентичност (Federated Identity) разчита на делегиране на на права на външен доставчик за удостоверяване на самоличността на потребителите. По този начин шаблонът постига сигурност на идентифицирането на потребителите, като същевременно опростява разработката. Свежда до минимум необходимостта от администрирането им и същевременно подобрява практическата работа на приложението.

Шаблонът Федеративна идентичност предлага решение, при което идентифицирането на потребителя, вече не е задача на приложението. Разчита се на делегиране на отговорността по автентикацията на потребителите на специализиран, сигурен доставчик на услугата по идентифициране. Той следва принципа за интегриране и координация с други контроли за сигурност на различни слоеве на облака така, че да се постигне цялостна защита на данните.

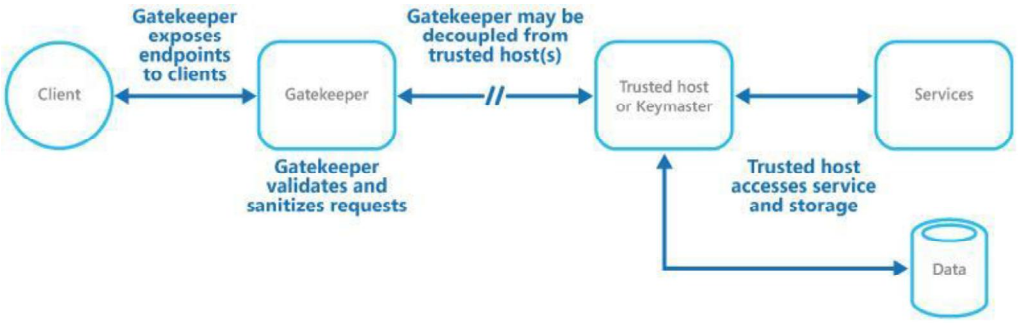
Използването на шаблона Федеративна идентичност е подходящо решение за клиентското приложение, което предлага достъп до услуга, изискваща идентифициране на ползвателите. Автентикацията се извършва от доставчик на услугата по идентифициране, който разпознава цифрови сертификати, издадени от нарочен сертификационен орган (Certificate Authority (CA)), като част от инфраструктура с публичен ключ. Цифровите сертификати предоставят информация за идентифициране на потребителя. Те съдържат информация за самоличността на потребителя, но могат да съдържат друга информация, като роля му, оторизациите и права за достъп. Използването на шаблона и издадените цифрови сертификати непрекъснато нараства и може да се наложи, като стандартен начин за удостоверяване на самоличност в Интернет.

Шаблонът Портиер (Gatekeeper Pattern) осигурява допълнително ниво на сигурност за използваните услуги или приложения в облака (фиг. 8), като така решава част от проблемите на архитектурата на микроуслугите. Той се явява посредник между клиента и услугата, валидира заявките, като предава само валидните заявки и данни. Предложеният от шаблона посредник има ограничена функционалност и не съхранява важна информация, като данни за достъп. Целта е да се минимизира риска клиентите да получат достъп до защитена информация или услуга, чрез разделяне на точките за достъп на потребителя от програмните компоненти, които изпълняват заявките и имат права за достъп до хранилищата на данните. Постига се с използването на шаблон Фасада или друг програмен модул, която взаимодейства с клиентите и след това препраща заявката например чрез отделен интерфейс

- към хостовете или задачите, които ще я обработват.



Фигура 7. Шаблон Федеративна идентичност.



Фигура 8. Шаблон Портиер.

Заключение

Изграждането на качествена и сигурна облачна система е предизвикателство пред разработчиците, което изисква познаването на проблемите и спазването на принципите на компютърната сигурност. В шаблоните са застъпени основните принципи на сигурността, затова те улесняват разработката, гарантират качеството на софтуера, позволяват бъдещето му развитие. Шаблоните подпомагат, както специалистите, създаващи облачната среда, разработчиците на криптографски протоколи и примитиви, а така също и софтуерните специалисти, които създават приложения, работещи в тази среда и предлагащи различни облачни услуги. Те сравнително лесно могат да се приложат, така че да се създаде система, отговаряща на изискванията за сигурност. Чрез шаблоните могат да се намалят както разходите, така и рисковете за сигурността.

Използвана литература

1. Гама, Е., Хелм. Р., Джонсън, Р., Влисидес, Дж., Шаблони за дизайн, СофтПрес, 2004.
2. Gamma, E. & Helm, R. & Johnson, R. & Vlissides, J., 1993. Design Patterns: Abstraction and Reuse of Object-Oriented Design. н.м.: Proceedings of ECOOP '93, pp. 406- 431.
3. Dara, S., 2014. Privacy Patterns in Public Clouds. н.м.: Proceedings of the Indian Conference

on Pattern Languages of Programs (GuruPLoP).
4. Dougherty, Ch., Sayre, K., Seacord, R., Svoboda, D., Togash, K., Secure Design Patterns, CERT Program report, Carnegie Mellon University, 2009.
5. Erl, T. & Cope, R. & Naserpour, A., 2015. Service-Oriented Architecture: Concepts, Technology, and Design. н.м.: Prentice Hall.
6. Fernandez, Larrondo-Petrie, Gudes, A method-based authorization model for objectoriented databases, Proc. of the OOPSLA 1993., Workshop on Security in Object-oriented Systems , 70-79.
7. Fernandez, Larrondo-Petrie, Teaching a course on data and network security using UML and patterns, Procs. of the Educators Symposium of MoDELS/UML 2005, Montego Bay, Jamaica, October 2-7, 2005.
8. Fernandez, E. B. Yoshioka, N. and Washizaki, H. "Patterns for Cloud Firewalls." In: Proceedings of the Asian Conference on Pattern Languages of Programs (AsianPLoP). 2014.
9. Hafiz, M., 2013. Security Pattern Catalog. <http://www.munawarhafiz.com/securitypatterncatalog/index.php>
10. Homer, A. & Sharp, J. & Brader, L. & Narumoto, M. & Swanson, Tr., 2014. Cloud Design Patterns. н.м.: Microsoft. 978-1-62114-036-8.
11. Hanmer, R., 2007. Patterns for Fault Tolerant Software. н.м.: Wiley.
12. Hanmer, R., 2013. Pattern-Oriented Software Architecture for Dummies For Dummies. н.м.: Wiley.
13. Hanmer, R., 2014. Patterns for Fault Tolerant Cloud Software. н.м.: Proceedings of the Conference on Pattern Languages of Programs (PLoP).
14. Romanosky S., Security Design Patterns, Technical report, 2001, <http://www.cgisecurity.com/lib/securityDesignPatterns.pdf>
15. Schumacher, M. & Fernandez, E. B. & Hybertson, D. & Buschmann, F. & Sommerlad, P., 2006. Security Patterns: Integrating Security and Systems Engineering. н.м.: John Wiley & Sons.
16. Yoder, J. & Barcalow, J., 1998. Architectural patterns for enabling application security. н.м.: Procs. PLOP'97.
17. <http://cloudpatterns.org/design_patterns>
18. <https://docs.microsoft.com/en-us/azure/>
19. <http://craigread.cloud/security-design-patterns-overview/>
20. <http://www.munawarhafiz.com/securitypatterncatalog/index.php>

За контакти
ас. д-р Мария Армянова
ИУ-Варна
armianova@ue-varna.bg