

## АННОТАЦИЯ

Тема выпускной квалификационной работы магистра — «Исследование проблем безопасности облачных технологий».

Ключевые слова: безопасность, облачные вычисления, виртуализация, уязвимости, защита информации, стандартизация.

В данной выпускной квалификационной работе магистра рассмотрены проблемы обеспечения безопасности облачных технологий. Для достижения поставленной цели был необходим детальный анализ мировых стандартов безопасности для облачных провайдеров, а также практический пентестинг облачных систем с целью эксплуатации наиболее опасных уязвимостей.

Актуальность темы. В настоящее время наблюдается стремительное развитие облачных технологий, однако для эффективной работы облачной инфраструктуры требуется эффективная структура и организация. Зачастую небольшая команда, проектирующая инфраструктуру облака не всегда может полностью учесть все аспекты безопасности, так как нет единого документа, стандартизирующего механизмы обеспечения безопасности. Особенно остро, вопрос безопасности встал в последнее время, в связи с обнаружением большого количества опасных уязвимостей в программном обеспечении, используемом для облаков.

Конечная цель проектирования — анализ проблем безопасности в облачных технологиях, а также практический пример эксплуатации опасной уязвимости в данной среде.

Выпускная квалификационная работа магистра изложена на ??? листах, включает ??? таблиц, ??? рисунков, ??? приложений, ??? литературных источников.

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ . . . . .	6
1 ПОСТАНОВКА ЗАДАЧИ . . . . .	7
2 ОБЗОР ЛИТЕРАТУРНЫХ ИСТОЧНИКОВ ПО ТЕМАТИКЕ ИССЛЕ- ДОВАНИЯ . . . . .	9
3 ОПИСАНИЕ РАБОТЫ, ПОКА ХЗ ЧТО ТУТ . . . . .	20
4 АНАЛИЗ ПОЛУЧЕННЫХ РЕЗУЛЬТАТОВ . . . . .	25
ЗАКЛЮЧЕНИЕ . . . . .	26
ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ . . . . .	27
БИБЛИОГРАФИЧЕСКИЙ СПИСОК . . . . .	28
СПИСОК ИЛЛЮСТРАТИВНОГО МАТЕРИАЛА . . . . .	29

## ВВЕДЕНИЕ

Облачные услуги — это способ предоставления, потребления и управления технологией. Данный тип услуг выводит гибкость и эффективность на новый уровень, путем эволюции способов управления, таких как непрерывность, безопасность, резервирование и самообслуживание, которые соединяют физическую и виртуальную среду.

Для эффективной работы облачной инфраструктуры требуется эффективная структура и организация. Небольшая команда из специалистов и бизнес-пользователей может создать обоснованный план и организовать свою работу в инфраструктуре. Данная выделенная группа может намного эффективнее построить и управлять нестандартной облачной инфраструктурой, чем если компании будут просто продолжать добавлять дополнительные сервера и сервисы для поддержки центра обработки данных (ЦОД).

Развитие информационного мира движется в сторону повсеместного пространства облачных вычислений, их технологий и сервисов. Очевидные преимущества данного подхода: [1]

- снижение затрат — отсутствие необходимости покупки собственного оборудования, программного обеспечения (ПО), работы системного инженера;
- удаленный доступ — возможность доступа к данным облака из любой точки мира, где есть доступ в глобальную сеть;
- отказоустойчивость и масштабируемость — изменение необходимых ресурсов в зависимости от потребности проекта, техническое обслуживание оборудования лежит на плечах облачного провайдера.

В связи с этим можно сделать вывод, что основные недостатки облачных вычислений сводятся к информационной безопасности. Такого мнения придерживаются многие крупные информационные компании, что в некоторой степени препятствует более стремительному развитию рынка облачных сервисов.

## 1 ПОСТАНОВКА ЗАДАЧИ

Конечной задачей выпускной квалификационной работы магистра на тему «Исследование проблем безопасности облачных вычислений» является подробный анализ стандартов безопасности облачных вычислений, варианты решения данных проблем, а также технические возможности практической эксплуатации уязвимостей на нескольких уровнях работы облачной инфраструктуры.

Исследования должны состоять из следующих частей:

- составные части облачной инфраструктуры;
- анализ технологий используемых облачными провайдерами, необходимых для построения облачной инфраструктуры;
- специфика применений облачных вычислений в России;
- проблемы безопасности облачных вычислений;
- решение проблем безопасности облаков;
- практическое применение уязвимостей в облачной среде, с использованием программ, распространяющихся под свободными лицензиями, например GNU GPL.

Для применения практических навыков исследования уязвимостей необходима аппаратная платформа со следующими характеристиками:

- процессор 4xIntel Core® i3 @ 2.3GHz с поддержкой аппаратной виртуализации;
- минимальный объем ОЗУ 8 Гб, рекомендуемый — не менее 10 Гб;
- минимум 15 Гб места на жестком диске (SSD);
- операционная система Ubuntu 16.04, CentOS 7 или Debian 8 GNU/Linux.

Данная задача также рассматривается с точки зрения системного и вариантного анализа.

Системный анализ включает в себя: [2]

- системотехническое представление системы безопасности в виде «черного ящика»;

- описание входных и выходных данных;
- список функций, которые выполняет система безопасности;
- учет случайностей;
- декомпозицию системы и описание связей между ее элементами.

Вариантный анализ произведен исходя выбранных критериев: [3]

- раз;
- два;
- три;
- ...;
- последний критерий.

## 2 ОБЗОР ЛИТЕРАТУРНЫХ ИСТОЧНИКОВ ПО ТЕМАТИКЕ ИССЛЕДОВАНИЯ

Исторически, ситуация сложилась так, что слово «облако» используется в качестве метафоры сети Интернет. Позже, оно было использовано для изображения Интернет в компьютерных сетевых диаграммах и схемах.

Облачные вычисления можно обозначить, как выделение ресурсов в облаке. В соответствии с NIST (Национальный институт стандартов и технологий), формальное определение облачных вычислений заключается в следующем: «Облачные вычисления являются моделью обеспечения повсеместного, удобного доступа по требованию по сети, общему пулу конфигурируемых вычислительных ресурсов (например сетей, серверов, систем хранения данных (СХД), приложений и услуг), которые могут быстро и с минимальными усилиями предоставлены для управления поставщиком услуг». [4]

Согласно опросам института Понемона в 2016 году, среди 3476 респондентов в сфере информационной безопасности из Соединенных Штатов Америки, Великобритании, Австралии, Германии, Японии, Франции, Японии, России, Индии и Бразилии, 73% респондентов так или иначе используют облачные вычисления в своей инфраструктуре. Особый рост внедрения облачных сервисов произошел в последние 2 года. [5]

Хранение данных пользователей, почты и потребительских данных в облаке выросло в 2016 году по сравнению с 2014 годом.

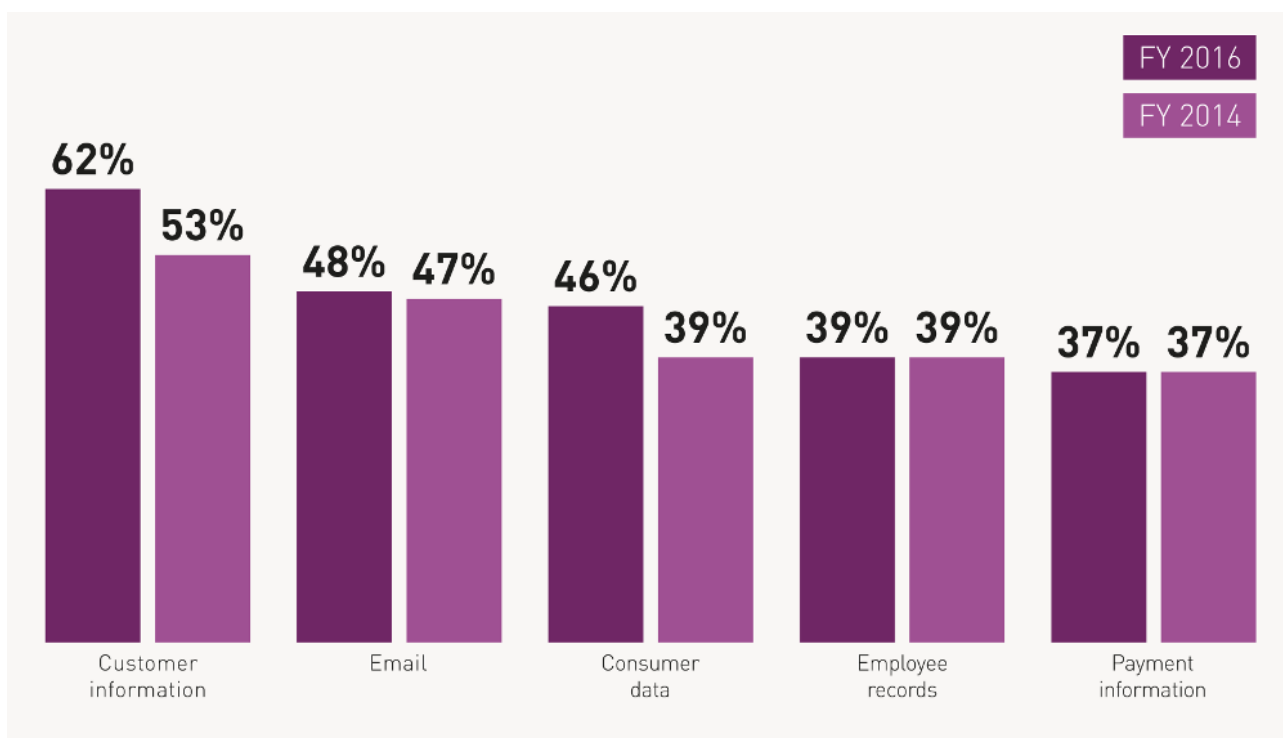


Рисунок 2.1 – Сравнение использования облачных вычислений для хранения данных

Облачные вычисления являются результатом объединения большого количества технологий и связующего ПО для обеспечения ресурсов, необходимых для решения задачи, балансировки процессов, мониторинга, автоматизации и прочего.

Основными отличиями облачного сервиса от классического являются:

- виртуализация;
- оркестратор;
- список услуг;
- портал самообслуживания;
- система тарификации и выставления счетов (биллинг).

В вычислениях, виртуализация является процессом создания виртуальной (не физической) версии чего-либо, в том числе аппаратных платформ виртуального компьютера, операционных систем, устройств хранения данных и вычислительных ресурсов.

Виртуализация может быть предоставлена на различных аппаратных и программных уровнях, таких как центральный процессор, диск, память, фай-

ловые системы и прочее. Чаще всего виртуализация используется для создания виртуальных машин и эмуляции различного оборудования для последующей установки операционных систем (ОС) на них.

Виртуальные машины создаются на основе гипервизора, который работает поверх операционной системы хост-компьютера (физического компьютера, не виртуального). С помощью гипервизора возможна эмуляция аппаратных средств, таких как процессор, диск, сеть, память, а также установка гостевых операционных систем на них. Возможно создание нескольких гостевых виртуальных машин с различными операционными системами на гипервизоре. Например, можно взять машину на Linux и установить ее на «голое» железо (bare-metal), и после настройки гипервизора возможно создание нескольких гостевых машин на Linux и Windows.

На данный момент все современные процессоры поддерживают аппаратную виртуализацию, это необходимо для безопасного и эффективного обмена ресурсами между хост-системой и гостевыми системами. Большинство современных процессоров и гипервизоров также поддерживают вложенную виртуализацию, что позволяет создавать виртуальные машины внутри виртуальных машин.

Оркестратор является механизмом, выполняющий набор заданных операций по шаблону. В сервис-ориентированной архитектуре (SOA), оркестровка сервисов реализуется согласно стандарту BPEL (Business Process Execution Language). Это позволяет автоматизировать процессы создания услуг клиентов в облачной среде.

Список услуг как предоставляется пользователю в виде шаблонов готовых тарифов на портале самообслуживания, однако существуют и так называемые «конфигураторы», которые позволяют пользователю создать шаблон индивидуально.

Портал самообслуживания является инструментом, с которым работает непосредственно пользователь. Именно на портале обслуживания размещается список услуг, доступных клиенту.



Система тарификации и выставления счетов является необходимым механизмом для определения финансовых затрат пользователя в соответствии с затраченными ресурсами клиента.

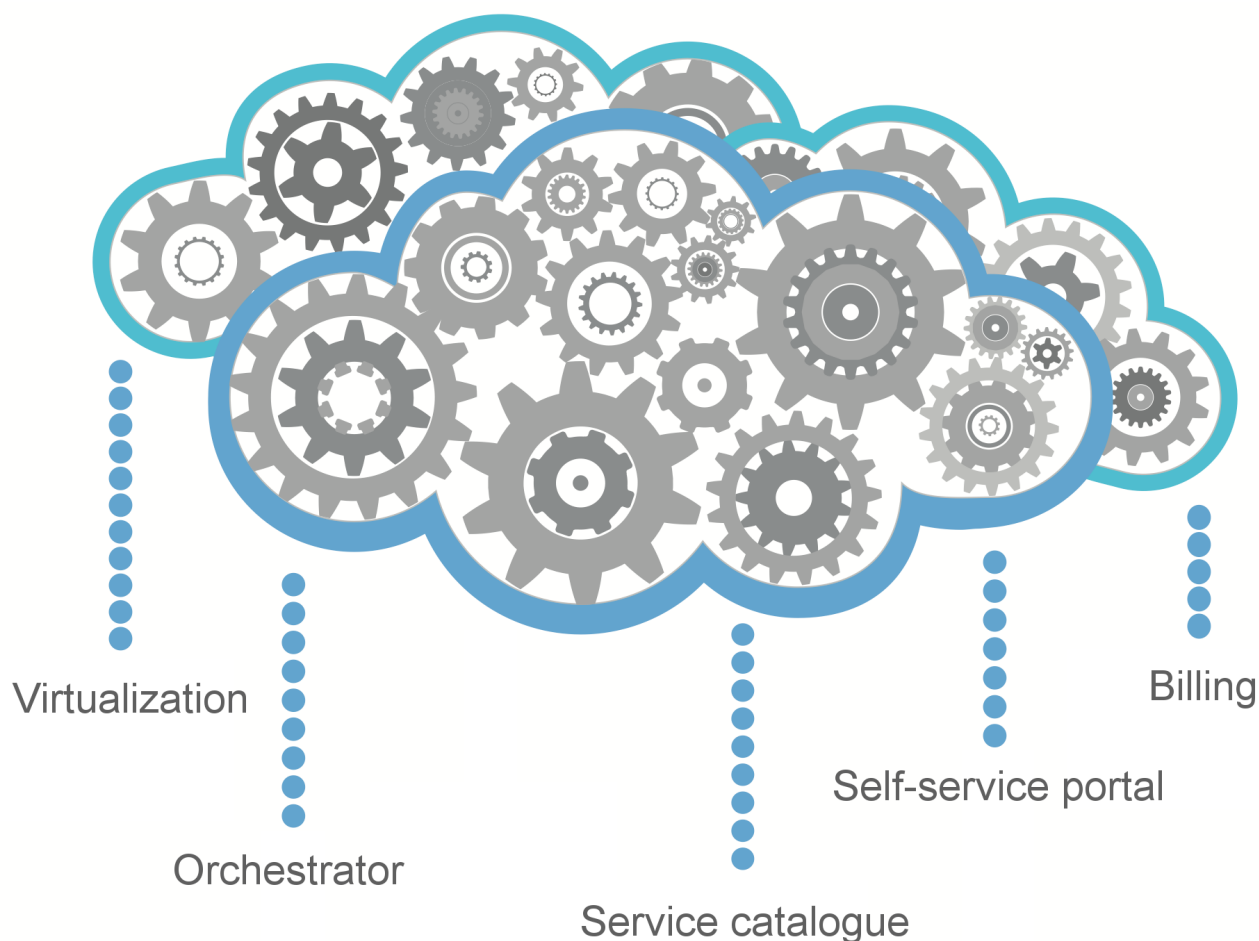


Рисунок 2.2 – Составные части облачных вычислений

Провайдеры облачных услуг предлагают различные виды услуг, построенных поверх базового резервирования и освобождения ресурсов. Большинство из этих услуг попадают в одну из следующих категорий:

- инфраструктура как услуга (IaaS);
- платформа как услуга (PaaS);
- программное обеспечение как услуга (SaaS).

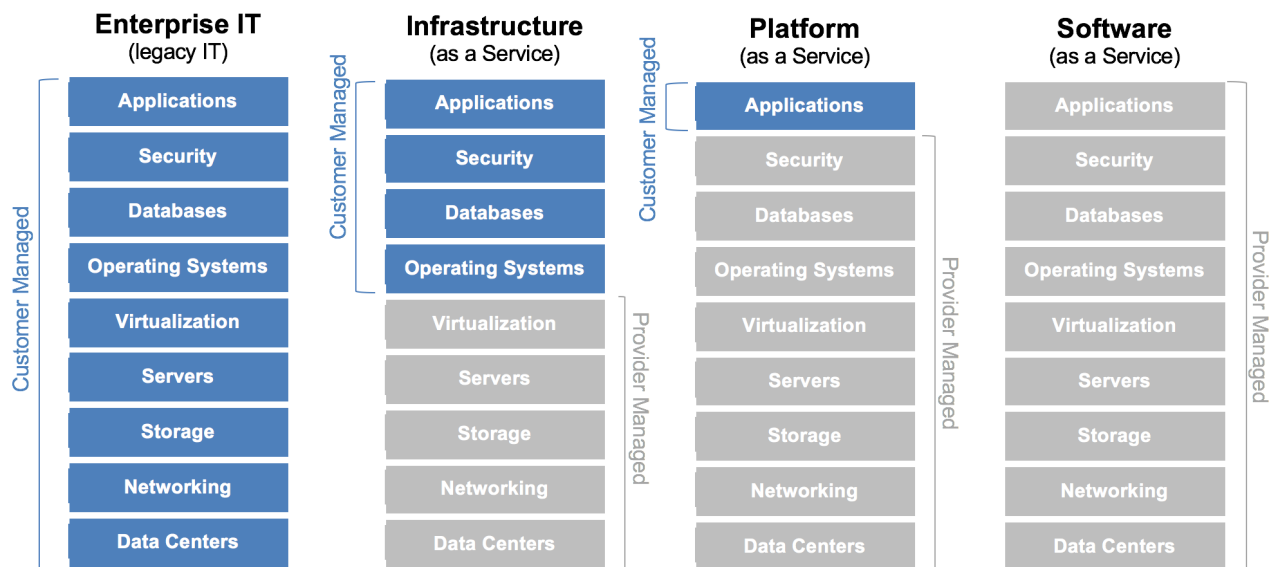


Рисунок 2.3 – Модели обслуживания облака

Большинство провайдеров используют различные виды веб-интерфейса, на основе которого можно построить необходимый стек технологий. Облачные провайдеры используют модель «pay-as-you-go», в которой оплата производится только за время использования ресурсов.

Ключевыми функциями облачных вычислений являются:

- скорость и масштабируемость, доступ к необходимым ресурсам можно получить одним щелчком мыши, что экономит время и обеспечивает гибкость, в зависимости от потребностей сервиса, можно легко масштабировать ресурсы как вверх, так и вниз;
- стоимость, снижение первоначальных затрат на развертывание инфраструктуры позволяет сосредоточиться на приложениях и бизнесе, облачные провайдеры имеют возможность заранее оценить стоимость, что значительно облегчает планирование бюджета;
- легкий доступ к ресурсам, пользователи могут получить доступ к инфраструктуре из любого места и устройства, до тех пор, пока существует подключение к провайдеру;
- обслуживание, все работы по техническому обслуживанию ресурсов осуществляются поставщиком облачных услуг, пользователи не должны беспокоиться об этом;

- мультиаренда, несколько пользователей могут использовать один и тот же пул доступных ресурсов;
- надежность, ресурсы могут быть размещены в разных дата-центрах, для обеспечения повышенной надежности.

Как правило, облако может быть развернуто согласно следующим моделям:

- частное облако;
- публичное облако;
- гибридное облако.

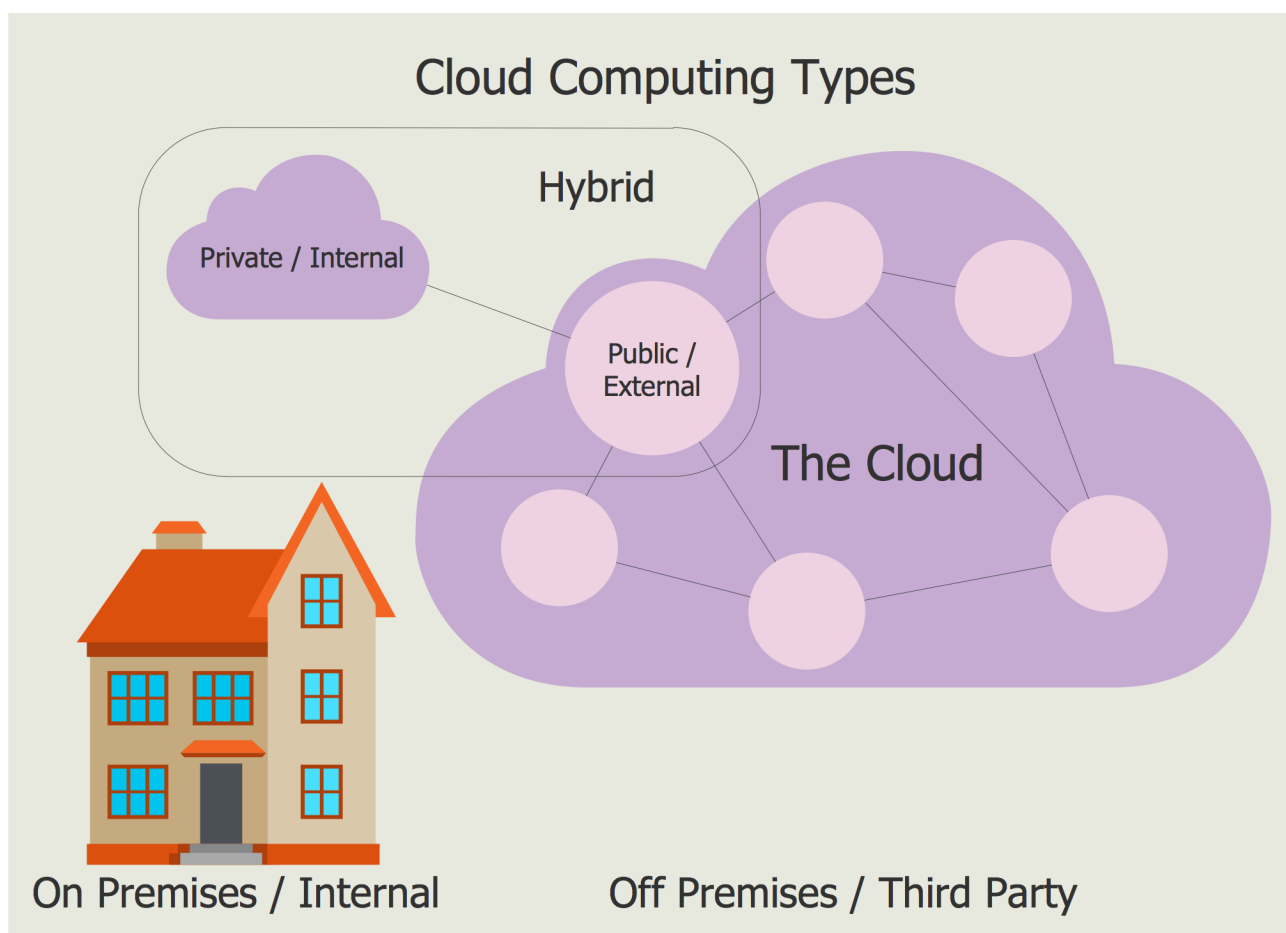


Рисунок 2.4 – Модели развертывания облака

Частное облако, эксплуатируется исключительно одной организацией, оно может быть размещено внутри или снаружи сети организации и управляться внутренними командами или третьей стороной. Частное облако можно построить с использованием такого программного обеспечения, как OpenStack;

Публичное облако доступно для всех пользователей, любой может использовать его после предоставления данных кредитной карты. AWS (Amazon Web Services) и GCE (Google Compute Engine) являются примерами публичных облаков;

Гибридное облако, является результатом объединения публичного и частных облаков. Гибридное облако может быть использовано для хранения секретной информации о частном облаке, предлагая при этом услуги на основе этой информации из публичного облака.

—

Актуальность: облака везде, облака нужны всем, не только бизнес-клиентам, но и обычным людям.

Т.к. популярность облаков появилась сравнительно недавно и она стремительно развивается, не всегда успевают учесть все аспекты безопасности.

Также из-за того, что облако состоит из большого количества ПО на различных уровнях, нужно учитывать все уязвимости, так как они могут всплыть на каждом из уровней.

Мысли:

- \* клиенты слабо представляют насколько защищены облака, поэтому предпочитают частное облаков, взамен публичного, не доверяют провайдеру

- \* основные аспекты облаков: мониторинг, управление, безопасность, доступность

- \* если надо добавить часть по экономике - файл 124.pdf

- \* по поводу иаас, преимущества понятны, а вот минусы в том, что если получают доступ к хост-ноде, то все, также это может быть изнутри, например уязвимость на гипервизоре

- \* конкретные проблемы с табличками описаны тут 1608.08787v1.pdf

- \* файл cloud-security-study-report.pdf конкретный отчет сравнение использования облаков в 2016 году по сравнению с 2014

- \* в файле informatsionnaya-bezopasnost-pri-oblachnyh-vychisleniyah-problemy-i-perspektivy.pdf хорошо расписан вопрос по стандартизации облаков, также кратко написано про риски использования облаков

- \* тут тоже про стандарты psta2011-4-17-31.pdf

- \* в файле str50.pdf рассказывается про проблемы в России, проблемы с точки зрения инф. безопасности

- \* реальные опросы от интела 2012г, которые рассказывают, что препятствуют уходу в облака, файл whats-holding-back-the-cloud-peer-research-report.pdf

- \* хорошая презентация Zegzhda-PD-supernova-2.pdf краткие тезисы, примеры картинок, модель безопасности даже есть

—



Рисунок 2.5 – Тестовая картиночка



Рисунок 2.6 – Тестовая картиночка

- + что такое облака
- + сравнение облаков 2014/16 тут графики
- + технологии облачных вычислений soa/asp/virt... тут картинки
- + saas/paas/iaas тут картинка
- + гибридное/публ/прив облако тут картинка
- \* стандартизация nist... тут можно табличку
- \* облачные провайдеры тут табличка/картинка
- \* специфика облаков в России

- \* тенденции развития облаков в мире (зеленые цоды, уход в виртуализированные хранилища и сети) пример картинки какой-то

- \* кратко по безопасности, основные пункты, кем регламентируется, как обеспечивается

Тут 10-20 страниц, без подпунктов, сплошной текст с картиночками, в общем теория.

- \* конкретный упор на безопасность уже в описании работы



### 3 ОПИСАНИЕ РАБОТЫ, ПОКА ХЗ ЧТО ТУТ

Пока я только знаю что тут примерно 20 страниц должно быть. Тут само исследование работы.

Все это должно быть с подпунктами.

почему именно linux и эти платформы? нужно собрать статистику использования дистрибутивов на серверах, а также платформ виртуализации возможно конкретно по россии эту цифру найти?

Уязвимости 2016, это нужно эксплуатировать

- \* Linux kernel, CentOS/RHEL/buntu/Debian Dirty COW

- \* Виртуализация: KVM/Xen/LXC/OpenVZ/Virtuozzo/VMWare/Hyper-V

- \* Протоколы: NTP/SSL/TLS/HTTP2 DROWN/POODLE/HEARTBLEED  
TCP overflow libc GHOST

- \* mysql cve-2016-3477

чтобы не ребутаться использовать kernelcare/kpatch, постоянно мониторинг уязвимостей

использовать lts дистрибутивы для поддержки софта

пример уязвимостей по kvm: <http://www.cvedetails.com/>

ntp - только на винде позволяет устроить ддос

есть drown до этого еще pooodle и heartbleed, это все уязвимости openssl, после этого появился libressl - она сложна в реализации <http://www.opennet.ru/opennews/art.shtml?num=43971>

нашумевшие еще это ghost, shellshock, CVE-2016-6663 (mysql) — конкретно

+ Linux kernel Dirty COW: CVE-2016-5195  
<http://www.opennet.ru/opennews/art.shtml?num=45354> получить рута можно

- Xen CVE-2016-6258 <http://www.opennet.ru/opennews/art.shtml?num=44855>  
выполнение произвольного кода на хост-ноде нет эксплоитов

- TCP CVE-2016-5696 <http://www.opennet.ru/opennews/art.shtml?num=44945>  
возможность обрыва tcp-соединения и подстановки данных в трафик тяжело эксплуатировать

- glibc CVE-2015-7547 <http://opennet.ru/opennews/art.shtml?num=43886> ее  
вряд ли можно эксплуатировать

- xen CVE-2016-3710 <http://www.opennet.ru/opennews/art.shtml?num=44409>  
работает только в hvm, выполнение кода на хост-ноде из гостевой

- linux kernel CVE-2016-8655 <http://www.opennet.ru/opennews/art.shtml?num=45573>  
тут возможно выйти за пределы lxc скорее всего, но тут сложно, для этого  
нужен CAP\_NET\_RAW и sysctl kernel.unprivileged\_userns\_clone=1 плюс нет  
эксплоита для LXC

- LXC CVE-2016-8649 <https://www.opennet.ru/opennews/art.shtml?num=45573>  
эксплоит есть, но у меня не работает на 11 строчке

—

Тестовый стенд для всего этого. лучше конечно это был бы дедик, но на  
крайняк kvm + nestedV

если все это можно эксплуатировать, то что делать?

\* мониторинг

\* если это что-то ядерное, то надо ребутать, так не пойдет, нужны патчи  
налету

\* использование встроенных механизмов защиты selinux/apparmor

\* если не удастся исправить онлайн патчем, то либо ребут (что критично),  
либо онлайн миграция

если связать это с опенсорсом, то

1. все источники уязвимостей из открытых данных  
2. эксплуатация уязвимостей тоже осуществляется с помощью свободно-  
го ПО

3. если это мониторинг, то скорее всего он тоже опенсорсный, но надо  
поискать если ли коммерческие альтернативы

Поискать скрипты, которые могут по открытым базам чекать уязвимости.  
Эту возможность можно запихнуть в мониторинг.

Дальше. Обзор наших провайдеров. Какие требования к ним предъявляются и как они их выполняют. Тут надо собрать статую по популярным облачным ребятам, почитать SLA и триальную услугу попробовать.

Если же речь идет не только об уязвимостях, но например еще о ддос, то тут помимо очевидного варианта атаки на инфраструктуру может быть, что в облаке клиента может быть зараза и это исходящий ддос. Это надо тоже как-то мониторить, решать это можно либо заблокировав клиента, либо если это легитимный трафик, то что-то делать с сетью.

Также в безопасность входят бекапы, фейловеры, все что соответствует SLA. Тут возможно сделать что-то с SDN и SDS, надо почитать.

ЕТО CHISTIY ISO CENTOS 7.2

```
[root@master ~]# cat /etc/redhat-release
CentOS Linux release 7.2.1511 (Core)
[root@master ~]# uname -r
3.10.0-327.el7.x86_64
[root@master ~]# su dcow
[dcow@master ~]$ id
uid=1000(dcow) gid=1000(dcow) groups=1000(dcow) context=
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[dcow@master ~]$ git clone https://github.com/gbonacini/CVE
-2016-5195.git
[dcow@master ~]$ cd CVE-2016-5195/
[dcow@master CVE-2016-5195]$ make
g++ -Wall -pedantic -O2 -std=c++11 -pthread -o dcow dcow.cpp -lutil
[dcow@master CVE-2016-5195]$ ./dcow
Running ...
Received su prompt (Password: )
Root password is: dirtyCowFun
Enjoy! :-)
[dcow@master CVE-2016-5195]$ su root
Password: dirtyCowFun
[root@master ~]# id
uid=0(root) gid=0(root) groups=0(root) context=unconfined_u:
unconfined_r:unconfined_t:s0-s0:c0.c1023
```

```

[root@master ~]# rpm -i http://patches.kernelcare.com/kernelcare-
latest.el6.x86_64.rpm
[root@master ~]# /usr/bin/kcarectl --info
kpatch-state: patch is applied
kpatch-for: Linux version 3.10.0-327.el7.x86_64 (builder@kbuilder.
dev.centos.org) (gcc version 4.8.3 20140911 (Red Hat 4.8.3-9) (
GCC) ) #1 SMP Thu Nov 19 22:10:57 UTC 2015
kpatch-build-time: Mon Nov 7 17:08:08 2016
kpatch-description: 20;3.10.0-327.36.3.el7.x86_64
[root@master ~]# /usr/bin/kcarectl --update
Kernel is safe
[root@master ~]# /usr/bin/kcarectl --uname
3.10.0-327.36.3.el7.x86_64
[root@master ~]# /usr/bin/kcarectl --patch-info | grep CVE
-2016-5195 -A3 -B3
kpatch-name: 3.10.0/0001-mm-remove-gup_flags-FOLL_WRITE-games-from-
__get_user-327.patch
kpatch-description: mm: remove gup_flags FOLL_WRITE games from
__get_user_pages()
kpatch-kernel: >kernel-3.10.0-327.36.2.el7
kpatch-cve: CVE-2016-5195
kpatch-cvss: 6.9
kpatch-cve-url: https://access.redhat.com/security/cve/cve
-2016-5195
kpatch-patch-url: https://git.kernel.org/linus/19
be0eaffa3ac7d8eb6784ad9bdbbc7d67ed8e619

[root@master ~]# uname -r
3.10.0-327.el7.x86_64

```

PROVEROCHKA

```
[dcow@master CVE-2016-5195]$ ./dcow
```

Running ...

NE RABOTAET

4\$/YEAR ZA 1 LICENZIUY

OTKLUYCHAEM

```
[root@master ~]# /usr/bin/kcarectl --unload
```

Updates already downloaded

KernelCare protection disabled, kernel might not be safe

```
[root@master ~]# su - dcow
```

Last login: Wed Dec 7 17:18:42 MSK 2016 on pts/0

```
[dcow@master ~]$ cd CVE-2016-5195/
```

```
[dcow@master CVE-2016-5195]$ ./dcow
```

Running ...

Received su prompt (Password: )

Root password is: dirtyCowFun

Enjoy! :-)

#### 4 АНАЛИЗ ПОЛУЧЕННЫХ РЕЗУЛЬТАТОВ

Тут подводим итог работы.

2 страницы выходит

## ЗАКЛЮЧЕНИЕ

Тут все понятно.

1 страница

## ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

- ЦОД — центр обработки данных
- ПО — программное обеспечение
- GNU — проект по разработке свободного программного обеспечения
- GPL — General Public License, универсальная общественная лицензия
- ОЗУ — оперативное запоминающее устройство
- SSD — Solid State Drive, твердотельный накопитель
- NIST — National Institute of Standards and Technology, Национальный институт стандартов и технологий
- СХД — система хранения данных
- SaaS — Software as a Service, программное обеспечение как услуга
- PaaS — Platform as a Service, платформа как услуга
- IaaS — Infrastructure as a Service, инфраструктура как услуга
- AWS — Amazon Web Services
- GCE — Google Compute Engine
- ОС — операционная система
- SOA — service-oriented architecture, сервис-ориентированная архитектура
- BPEL — Business Process Execution Language, язык описания бизнес-процессов



## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Прудникова, А.А. Безопасность облачных вычислений / А.А. Прудникова // Мир телекома. – 2013. - №1. – С.50-55.
2. Методические указания «Процедура системного анализа при проектировании программных систем» для студентов-дипломников дневной и заочной формы обучения специальности 7.091501 / Сост.: Сергеев Г.Г., Скатков А.В., Мащенко Е.Н. – Севастополь: Изд-во СевНТУ, 2005. – 32с.
3. Методические указания к расчетно-графическому заданию на тему «Метод анализа иерархий» по дисциплине «Теория оптимальных решений» для студентов специальности 7.091501 «Компьютерные системы и сети» дневной и заочной формы обучения / Сост.: Ю.Н. Щепин – Севастополь: Изд-во СевНТУ, 2008. – 28с.
4. Hogan, M., Liu, F., Sokol, A., Tong, J. NIST Cloud Computing Standarts Roadmap / NIST Special Publication 500-291, Version 2 Roadmap Working Group, 2013. – 113p.
5. The 2016 Global Cloud Data Security Study. Ponemon Insitute LLC, 2016. – 40p.

## СПИСОК ИЛЛЮСТРАТИВНОГО МАТЕРИАЛА

Картинки

1 страница