

# Выбор стека технологий вычислительной инфраструктуры для экспериментальных исследований цифровых валют

А.С. Албычев, Д.Ю. Ильин

**Аннотация**—Одним из направлений исследований цифровых валют центральных банков является экспериментальная оценка характеристик технологического обеспечения. Как правило, для реализации цифровых валют рассматриваются блокчейн-технологии. Оценка технических аспектов требует подготовки экспериментального стенда, для построения которого могут применяться виртуальные машины. Ввиду децентрализованного характера технологического обеспечения их количество может быть достаточно большим, поэтому для подготовки стенда необходимо выбрать адекватный задаче стек технологий вычислительной инфраструктуры. В работе рассматривается 2 варианта инфраструктуры для размещения стенда из 33 программно-управляемых виртуальных машин. На виртуальные машины производится установка Hyperledger Fabric, монитора вычислительных ресурсов atop и программного обеспечения для поддержки функционирования стенда и блокчейн-системы. Проведен экспериментальный запуск стенда на основе 2 наборов технологий вычислительной инфраструктуры. Стек из Windows Server и VirtualBox 7.1 не позволил осуществить запуск стенда на представленном оборудовании ввиду избыточного и непропорционального использования ресурсов центрального процессора. В свою очередь, стек технологий с гипервизором VMware ESXi 7.0 позволил осуществить успешный запуск экспериментального стенда, однако он требует дополнительной настройки сервисных виртуальных машин для функционирования вычислительной инфраструктуры.

**Ключевые слова**—Цифровые валюты центральных банков, блокчейн, вычислительная инфраструктура, экспериментальный стенд.

## I. ВВЕДЕНИЕ

Цифровые валюты – сравнительно новый финансовый инструмент. Как отмечается в [1], нет какой-то единой и устоявшейся классификации, однако, исследователи выделяют в отдельные группы частные цифровые валюты (криптовалюты), такие как Bitcoin, Ethereum, и цифровые валюты центральных банков (central bank digital currency, CBDC). Для большей части цифровых валют характерно использование технологии

распределенного реестра [2], одним из основных примеров которой является технология блокчейн.

В исследованиях CBDC рассматриваются вопросы того, следует ли центральным банкам выпускать цифровые валюты [3], какими должны быть свойства цифровой валюты [4]–[6], и то, какие технологии [7] должны использоваться в основе цифровых валют центрального банка.

Во многих экспериментальных исследованиях технологического обеспечения CBDC, так же, как и для частных цифровых валют, используются блокчейн-технологии. Для этих целей часто выбирают гибкие и настраиваемые платформы, такие как Corda, Quorum или Hyperledger Fabric. Они фигурируют в целом ряде [8] экспериментов.

Алгоритмы консенсуса являются неотъемлемой частью исследований CBDC на основе блокчейн-технологий [2] ввиду распределенного характера системы. Их применение становится целесообразным при размещении технологического обеспечения у нескольких операторов CBDC.

Достижение консенсуса в распределенных вычислениях может быть ресурсоемким процессом [9], [10]. Независимо от формы представления цифровой валюты, при проектировании следует определять приемлемое число операторов CBDC, участвующих в достижении консенсуса, так как это определяет баланс [11] между производительностью и защищенностью CBDC.

Hyperledger Fabric предоставляет инструментарий для настройки правил [9], необходимых для одобрения транзакции и последующего достижения консенсуса, поэтому эту платформу можно использовать для оценки ресурсоемкости обработки транзакций при разных параметрах блокчейн-сети.

Для этого нужно организовать экспериментальные исследования с достаточным числом организаций. Каждая организация может быть представлена набором виртуальных машин (ВМ) [10], на которых будут размещены узлы системы.

Чтобы обеспечить работу экспериментального стенда необходимо определить стек технологий для построения подходящей вычислительной инфраструктуры. Она должна обеспечивать эффективную работу заданного числа ВМ и корректную работу средств для сбора данных [12] об использовании вычислительных

Статья получена 09 марта 2023.

А.С. Албычев, Федеральное казначейство Министерства финансов Российской Федерации, МИРЭА — Российский технологический университет, Москва, Россия (e-mail: albychev@mirea.ru).

Д.Ю. Ильин, МИРЭА – Российский технологический университет, Москва, Россия (e-mail: i@dmitryilin.com).

ресурсов.

Выбранный стек технологий (операционная система [13], гипервизор [13], [14], средства сбора данных мониторинга [15], [16]) может оказывать значительное влияние на производительность и надежность работы вычислительной инфраструктуры. Поэтому цель настоящей работы – выбрать стек технологий вычислительной инфраструктуры, применимой для проведения экспериментальных исследований цифровых валют центрального банка.

Далее работа имеет следующую структуру. В разделе «конфигурация программно-управляемого экспериментального стенда» описаны группы узлов экспериментального стенда и приведен перечень программного обеспечения, устанавливаемого на них. Далее, в разделе «конфигурации вычислительной инфраструктуры» указаны количественные параметры экспериментального стенда, используемого для оценки инфраструктур и два стека технологий, рассмотренные для постановки экспериментальных исследований. В разделе «анализ результатов» обсуждаются сделанные в ходе построения и запуска инфраструктур наблюдения. В заключении подведены итоги эксперимента и приведены выводы.

## II. КОНФИГУРАЦИЯ ПРОГРАММНО-УПРАВЛЯЕМОГО ЭКСПЕРИМЕНТАЛЬНОГО СТЕНДА

### A. Узлы экспериментального стенда

Для работы экспериментального стенда с Hyperledger Fabric необходимы следующие категории узлов: Peer-узлы, Orderer-узлы, узлы Certification Authority (CA) и узел Client.

Peer-узлы – это узлы, на которых осуществляются основные вычисления. Они хранят данные, обрабатывают поступающие транзакции и исполняют чейнкод. Также они являются точкой подключения для узла Client и начиная с Hyperledger Fabric версии 2.4, выполняют роль Gateway-сервиса. Эта роль означает, что Peer-узел осуществляет процедуру проверки и подтверждения результата транзакции без необходимости узла Client осуществлять отдельный запрос для каждого из этапов обработки транзакции.

Orderer-узлы отвечают за упорядочивание одобренных транзакций и формирование новых блоков в блокчейн-сети. Они рассылают новые блоки Peer-узлам для последующего добавления в цепочку блоков и хранения.

Узлы Certification Authority нужны для создания сертификатов, используемых для формирования защищенных подключений и разграничения прав доступа в рамках работы Hyperledger Fabric.

В рамках Hyperledger Fabric Peer-узлы и Orderer-узлы относятся к какой-либо организации, участвующей в блокчейн-сети. Каждой из организаций можно выделить отдельный узел Certification Authority. Общая схема экспериментального стенда представлена на рисунке 1. Подразумевается, что в экспериментальных целях может изменяться количество организаций, количество Peer-

узлов в организациях и количество Orderer-узлов.

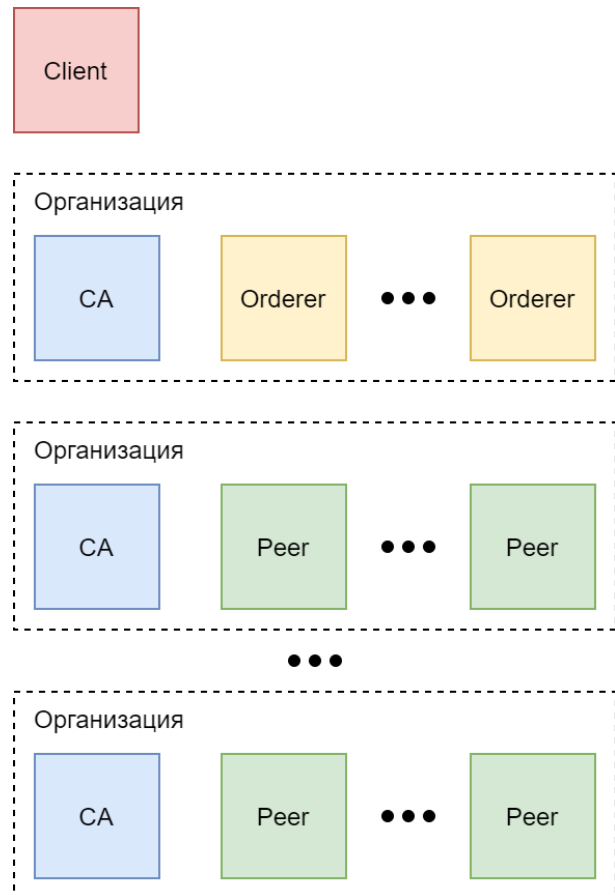


Рисунок 1 – Общая схема экспериментального стенда

### B. Программное обеспечение узлов экспериментального стенда

Для сбора данных об использовании вычислительных ресурсов каждым узлом экспериментального стенда, а также для изоляции выделенных ресурсов, каждый элемент стенда будет размещен в отдельной ВМ.

На каждую программно-управляемую ВМ устанавливается следующее программное обеспечение:

- Ansible – инструмент для автоматизации процесса установки и конфигурации [17] программного обеспечения;
- htop, iotop – мониторы использования вычислительных ресурсов [18];
- atop – монитор использования вычислительных ресурсов [18], в том числе работающий в фоновом режиме и сохраняющий историю наблюдений;
- rsync – инструмент для синхронизации содержимого директорий, используемый в рамках стенда в том числе для обмена данными с ВМ;
- htpdate – утилита для синхронизации времени операционной системы по протоколам http и https;
- sshpass – утилита для ввода пароля при подключении по SSH в неинтерактивном режиме;
- Hyperledger Fabric – программное обеспечение рассматриваемой платформы, включающее как консольные утилиты, так и сервисы;

- Hyperledger Fabric CA – программное обеспечение, рекомендуемое разработчиками Hyperledger Fabric для управления сертификатами [19].

Кроме этого, на VM Peer-узлов производится установка:

- Docker – система контейнеризации [19], необходимая для исполнения чейнкода.
- Golang – средства языка Go [19], необходимые для установки зависимостей чейнкода и его запуска.

На VM узлов Certification Authority дополнительно производится установка:

- Docker – система контейнеризации [19], необходимая для запуска двух независимых Certification Authority узлов на одной VM (один для управления сертификатами, используемыми для разграничения прав доступа к блокчейн-сети, другой – для обеспечения защищенных подключений).
- Step CLI – утилита для оперирования инфраструктурой сертификатов

На VM Client устанавливается:

- NodeJS – интерпретатор языка JavaScript, используемый для запуска Hyperledger Caliper;
- Hyperledger Caliper – инструмент [20] для проведения нагрузочных испытаний для Hyperledger Fabric.

### III. КОНФИГУРАЦИИ ВЫЧИСЛИТЕЛЬНОЙ ИНФРАСТРУКТУРЫ

#### A. Экспериментальный стенд

Для того, чтобы определить стек технологий, применимый для реализации вычислительной инфраструктуры экспериментального стенда, необходимо задать конкретный перечень узлов.

В оценке инфраструктуры для Hyperledger Fabric будет использовано 7 организаций с 3 Peer-узлами в каждой. Также потребуется 1 организация с Orderer-узлами. Их количество целесообразно сделать равным 3, так как, исходя из документации, это минимальное количество узлов, при котором система сохранит функционирование при отключении одного из них. Каждой организации также понадобится 1 узел Certification Authority. Кроме этого, потребуется один узел Client, который впоследствии будет использован как генератор нагрузки.

Каждый узел целесообразно разместить в отдельной VM. Количественные параметры экспериментального стенда приведены в таблице 1.

ТАБЛИЦА 1 – СОСТАВ ВИРТУАЛЬНЫХ МАШИН ЭКСПЕРИМЕНТАЛЬНОГО СТЕНДА

Категория VM	Количество
Peer-узлы	21
Orderer-узлы	3
Узлы Certification Authority	8
Узлы Client	1
Всего	33

#### B. Инфраструктура на основе VirtualBox

Первый вариант инфраструктуры был подготовлен на выделенной VM с характеристиками, представленными в таблице 2.

ТАБЛИЦА 2 – ХАРАКТЕРИСТИКИ ХОСТ-СИСТЕМЫ ДЛЯ ЭКСПЕРИМЕНТАЛЬНОГО СТЕНДА НА ОСНОВЕ VIRTUALBOX

Элемент конфигурации	Характеристика
Тип процессора	Intel Xeon E5-2697 v4 @ 2.30ГГц
Виртуальные ядра процессора	70 (64 от сокета 0 и 6 от сокета 1)
Объем ОЗУ	74 Гб
Тип жесткого диска	HDD
Объем дискового пространства	500 Гб
Операционная система	Windows Server 2012 R2
Гипервизор	VirtualBox 7.1

В качестве инструмента для программного управления VM был использован Repexlab [21], в рамках которого используется стек инструментов Vagrant, Ansible, VirtualBox, atop. Общая схема конфигурации приведена на рисунке 2. Инструментарий управления экспериментом, так же, как и базовый образ (Vagrant Box) для создания VM, размещены на хост-системе. Под управлением программного обеспечения из базового образа создается набор VM, которые конфигурируются в соответствии с их ролями.

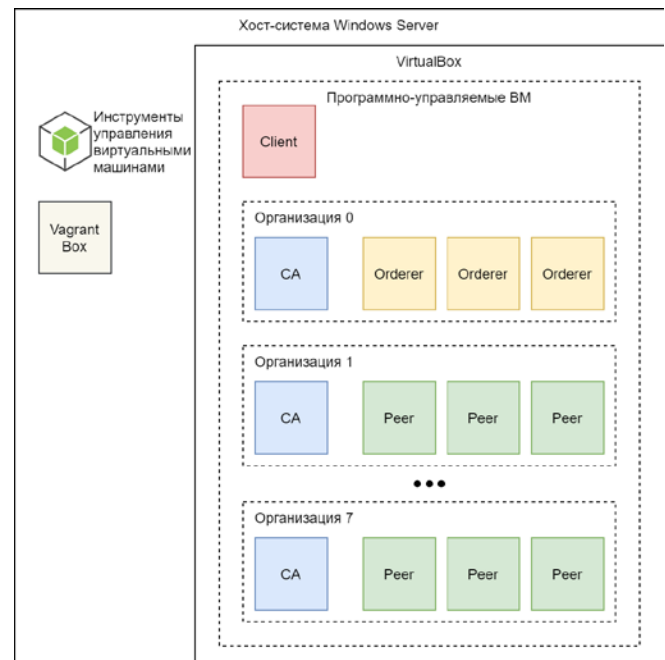


Рисунок 2 – Схема экспериментального стенда на основе VirtualBox

Конфигурация для всех программно-управляемых VM, за исключением VM Client, идентична и приведена в таблице 3.

ТАБЛИЦА 3 – ХАРАКТЕРИСТИКИ ПРОГРАММНО-УПРАВЛЯЕМЫХ ВИРТУАЛЬНЫХ МАШИН ЭКСПЕРИМЕНТАЛЬНОГО СТЕНДА НА VIRTUALBOX

Элемент конфигурации	Характеристика
Виртуальные ядра процессора	1 (4 для машины Client)
Объем ОЗУ	1 Гб (2 Гб для машины Client)
Объем дискового пространства	40 Гб, динамический (thin provisioned)
Операционная система	Ubuntu 20.04
Инструмент мониторинга ресурсов	atop (интервал – 1 секунда)

При создании и провизии 29 VM была обнаружена проблема, при которой нагрузка на ядра центрального процессора распределялась неравномерно. Скриншот на рисунке 3 иллюстрирует, что основной нагрузке были подвержены ядра процессора, относящиеся к сокету 1, тогда как сокет 0 практически не используется. При этом хост-система в целом становилась неотзывчивой и дальнейшее оперирование VM было невозможно.

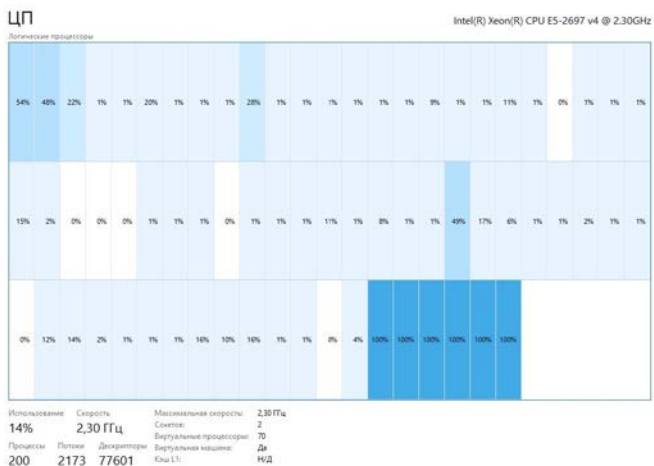


Рисунок 3 – Непропорциональное использования ресурсов двух сокетов ЦП после провизии VM

Схожая конфигурация экспериментального стенда была размещена на другой хост-системе (конфигурация в таблице 4) для исключения влияния конкретного оборудования и внешней инфраструктуры на эксперимент.

ТАБЛИЦА 4 – ХАРАКТЕРИСТИКИ ПРОВЕРОЧНОЙ ХОСТ-СИСТЕМЫ ДЛЯ ЭКСПЕРИМЕНТАЛЬНОГО СТЕНДА НА ОСНОВЕ VIRTUALBOX

Элемент конфигурации	Характеристика
Тип процессора	Intel Xeon Gold 6130 @ 2.10 ГГц
Виртуальные ядра процессора	32 (16 сокетов)
Объем ОЗУ	64 Гб
Тип жесткого диска	HDD
Объем дискового пространства	500 Гб
Операционная система	Windows Server 2022 Standard
Гипервизор	VirtualBox 7.1

При создании и провизии 25 VM была замечена схожая проблема потери отзывчивости хост-системы. При этом нагрузка на центральный процессор была значительна (рисунок 4) несмотря на распределение по всем ядрам.

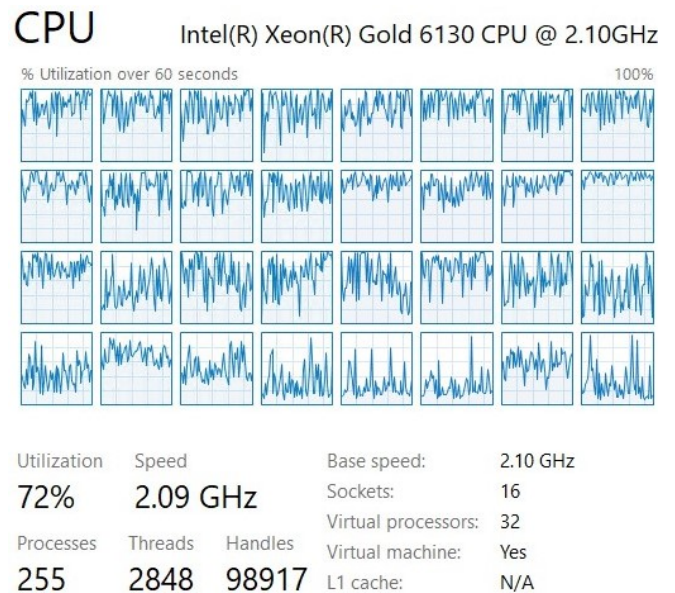


Рисунок 4 – Использование ресурсов ЦП после провизии VM при использовании VirtualBox

В связи с перечисленными проблемами инфраструктура на основе хост-системы Windows Server с гипервизором VirtualBox была оценена как неприменимая для проведения экспериментов с использованием 25 и более VM.

### С. Инфраструктура на основе VMWare ESXi

Второй вариант инфраструктуры был подготовлен с использованием VMWare ESXi с аналогичными характеристиками хост-системы, представленными в таблице 5.

ТАБЛИЦА 5 – ХАРАКТЕРИСТИКИ ХОСТ-СИСТЕМЫ ДЛЯ ЭКСПЕРИМЕНТАЛЬНОГО СТЕНДА НА ОСНОВЕ VMWARE ESXi

Элемент конфигурации	Характеристика
Тип процессора	Intel Xeon E5-2697 v4 @ 2.30ГГц
Виртуальные ядра процессора	70 (35 на сокет)
Объем ОЗУ	74 Гб
Тип жесткого диска	HDD
Объем дискового пространства	470 Гб
Гипервизор	ESXi 7.0 Update 3

В качестве инструмента для программного управления VM была использована модифицированная версия Rerexlab, адаптированная для работы с VMWare ESXi. Для этого были использованы плагины vagrant-vmware-esxi и утилита ovftool, обеспечивающие работу Vagrant с обозначенным гипервизором. Для провизии также использовалась система Ansible, а для мониторинга ресурсов VM – монитор atop. Общая схема

конфигурации приведена на рисунке 5.

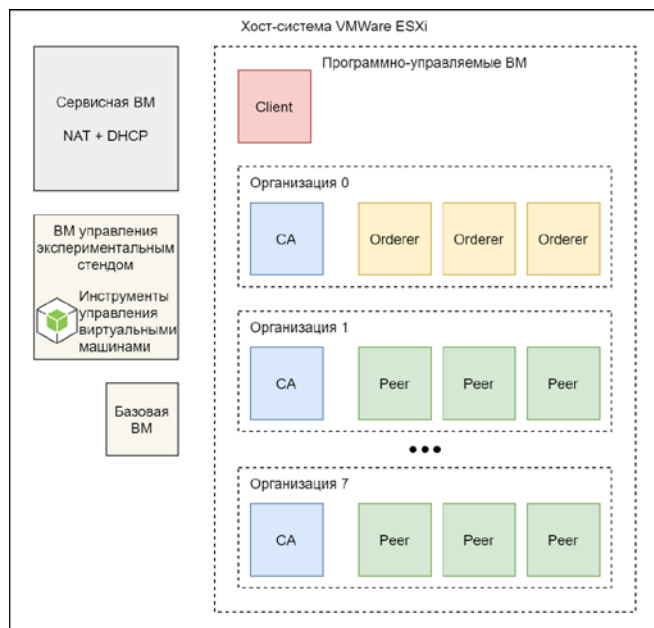


Рисунок 5 – Схема экспериментального стенда на основе VMware ESXi

Инструментарий управления экспериментом был размещен на выделенной ВМ, характеристики которой приведены в таблице 6.

ТАБЛИЦА 6 – ХАРАКТЕРИСТИКИ ВИРТУАЛЬНОЙ МАШИНЫ УПРАВЛЕНИЯ ЭКСПЕРИМЕНТАЛЬНЫМ СТЕНДОМ

Элемент конфигурации	Характеристика
Виртуальные ядра процессора	4
Объем ОЗУ	4 Гб
Объем дискового пространства	90 Гб, динамический (thin provisioned)
Операционная система	Ubuntu 22.04

Также для функционирования потребовалась ручная настройка сети NAT и сервиса DHCP с назначением IP-адресов в соответствии с MAC-адресами программно-управляемых ВМ. Это было выполнено с помощью сервисной ВМ, для которой были выделены ресурсы, приведенные в таблице 7.

ТАБЛИЦА 7 – ХАРАКТЕРИСТИКИ СЕРВИСНОЙ ВИРТУАЛЬНОЙ МАШИНЫ

Элемент конфигурации	Характеристика
Виртуальные ядра процессора	2
Объем ОЗУ	4 Гб
Объем дискового пространства	50 Гб, динамический (thin provisioned)
Операционная система	Windows Server 2016

Вместо базового образа ВМ использовалась заранее сконфигурированная ВМ (на рисунке 4 – «Базовая ВМ»), которая впоследствии клонировалась для создания и настройки программно-управляемых ВМ. Конфигурация программно-управляемых ВМ схожа с указанной ранее за исключением сокращенного до 10 Гб объема

дискового пространства (полная конфигурация приведена в таблице 8). Такое изменение связано с тем, что при значительной переподписке на дисковое пространство, даже при наличии фактического свободного места на диске, гипервизор VMware ESXi приостанавливал функционирование ВМ.

ТАБЛИЦА 8 – ХАРАКТЕРИСТИКИ ВИРТУАЛЬНЫХ МАШИН ЭКСПЕРИМЕНТАЛЬНОГО СТЕНДА

Элемент конфигурации	Характеристика
Виртуальные ядра процессора	1 (4 для машины Client)
Объем ОЗУ	1 Гб (2 Гб для машины Client)
Объем дискового пространства	10 Гб, динамический (thin provisioned)
Операционная система	Ubuntu 20.04
Инструмент мониторинга ресурсов	atop (интервал – 1 секунда)

При запуске и провизии всех 33 программно-управляемых ВМ нагрузка на ЦП оказалась приемлемой, а система не потеряла отзывчивости. Собранные данные статистики использования ресурсов хост-системы с интервалом 20 секунд показали, что нагрузка на ЦП составляла 14% с единичными пиками до 25% с периодичностью от 3 до 8 минут.

Так как потребление ресурсов ЦП может быть заметным при проведении, была проведена проверка влияния настроек atop на использование ресурсов ЦП хост-системы. Интервал сбора данных мониторинга был увеличен до 10 секунд, и отключен сбор данных об используемом процессами пропорциональном объеме памяти. В результате использование ресурсов ЦП хост-системы сократилось до 3.6% с единичными пиками до 11% с периодичностью от 3 до 8 минут.

#### IV. АНАЛИЗ РЕЗУЛЬТАТОВ

Был проведен запуск двух инфраструктур, рассматриваемых для проведения экспериментального исследования Hyperledger Fabric. Для инфраструктуры на основе гипервизора VirtualBox были получены неудовлетворительные результаты, не позволяющие использовать ее для проведения исследований для заданного количества ВМ. Инфраструктура на основе VMware ESXi потребовала внести модификации в используемый инструментарий, а также провести ручную настройку сети и привязку IP-адресов к ВМ. Тем не менее, данный гипервизор позволил провести успешный запуск заданного количества ВМ.

Ввиду высокого потребления ресурсов ЦП монитором ресурсов atop целесообразно увеличить интервал сбора информации. Также, как следует из документации atop, потребление ресурсов ЦП можно сократить, осуществляя запуск без флага «-R». Он отвечает за сбор данных об используемом процессами пропорциональном объеме памяти (proportional set size или PSIZE) и включен по умолчанию при установке atop на операционной системе Ubuntu 20.04. Благодаря этому

можно сократить долю ресурсов ЦП, используемых для мониторинга, однако сократится детализация полученных данных.

#### V. ЗАКЛЮЧЕНИЕ

В работе был рассмотрен вопрос выбора стека технологий вычислительной инфраструктуры, применимой для проведения экспериментальных исследований CBDC.

Определено, что для проведения экспериментальных исследований над технологическим обеспечением CBDC необходим стенд, размещенный в адекватной вычислительной инфраструктуре.

Приведено общее описание экспериментального стенда на основе программно-управляемых ВМ. Указаны 4 группы узлов экспериментального стенда: Orderer, Peer, CA, Client. Для каждой из групп приведен набор программного обеспечения, устанавливаемого на соответствующие ВМ.

Рассмотрены две инфраструктуры на разном наборе технологий. В результате экспериментального запуска стенда выявлено, что для заданного числа ВМ целесообразно использовать VMWare ESXi, тогда как связка хост-системы Windows Server и гипервизора VirtualBox не позволила организовать работу стенда.

В вычислительной инфраструктуре на основе Windows Server и VirtualBox наблюдается неравномерное и избыточное использование ресурсов центрального процессора. В дальнейших исследованиях следует рассмотреть работу гипервизора в сочетании с операционными системами Linux, чтобы определить, является ли наблюдаемый эффект специфичным для операционной системы или для гипервизора.

Полученные теоретические результаты могут быть применены для выбора стека технологий для построения иных экспериментальных стендов по исследованию CBDC.

Полученные практические результаты позволяют осуществлять экспериментальные исследования на подготовленной инфраструктуре со сбором данных об использовании вычислительных ресурсов каждым отдельным узлом.

#### БИБЛИОГРАФИЯ

- [1] M. N. Alsalmi, S. Ullah, and M. Rafique, "Accounting for Digital Currencies," *Research in International Business and Finance*, p. 101897, Feb. 2023, doi: 10.1016/j.ribaf.2023.101897.
- [2] N. Dashkevich, S. Counsell, and G. Destefanis, "Blockchain Application for Central Banks: A Systematic Mapping Study," *IEEE Access*, vol. 8, pp. 139918–139952, 2020, doi: 10.1109/ACCESS.2020.3012295.
- [3] T. Keister and D. Sanches, "Should Central Banks Issue Digital Currency?," *The Review of Economic Studies*, vol. 90, no. 1, pp. 404–431, Jan. 2023, doi: 10.1093/restud/rdac017.
- [4] I. Agur, A. Ari, and G. Dell'Ariccia, "Designing central bank digital currencies," *Journal of Monetary Economics*, vol. 125, pp. 62–79, Jan. 2022, doi: 10.1016/j.jmoneco.2021.05.002.
- [5] Y. Lee, B. Son, S. Park, J. Lee, and H. Jang, "A Survey on Security and Privacy in Blockchain-based Central Bank Digital Currencies," *Journal of Internet Services and Information Security*, vol. 11, no. 3, pp. 16–29, Aug. 2021, doi: 10.22667/JISIS.2021.08.31.016.
- [6] Y. Chu, J. Lee, S. Kim, H. Kim, Y. Yoon, and H. Chung, "Review of Offline Payment Function of CBDC Considering Security Requirements," *Applied Sciences*, vol. 12, no. 9, Art. no. 9, Jan. 2022, doi: 10.3390/app12094488.
- [7] T. Zhang and Z. Huang, "Blockchain and central bank digital currency," *ICT Express*, vol. 8, no. 2, pp. 264–270, Jun. 2022, doi: 10.1016/j.icte.2021.09.014.
- [8] E. A. Opare and K. Kim, "A Compendium of Practices for Central Bank Digital Currencies for Multinational Financial Infrastructures," *IEEE Access*, vol. 8, pp. 110810–110847, 2020, doi: 10.1109/ACCESS.2020.3001970.
- [9] H. Xiong, M. Chen, C. Wu, Y. Zhao, and W. Yi, "Research on Recent Progress of Blockchain Consensus Algorithms," *Future Internet*, vol. 14, no. 2, Art. no. 2, Feb. 2022, doi: 10.3390/fi14020047.
- [10] H. Javaid, C. Hu, and G. Brebner, "Optimizing Validation Phase of Hyperledger Fabric," in *2019 IEEE 27th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Oct. 2019, pp. 269–275. doi: 10.1109/MASCOTS.2019.00038.
- [11] S. Y. Jin and Y. Xia, "CEV Framework: A Central Bank Digital Currency Evaluation and Verification Framework With a Focus on Consensus Algorithms and Operating Architectures," *IEEE Access*, vol. 10, pp. 63698–63714, 2022, doi: 10.1109/ACCESS.2022.3183092.
- [12] J. Enes, R. R. Expósito, and J. Touriño, "BDWatchdog: Real-time monitoring and profiling of Big Data applications and frameworks," *Future Generation Computer Systems*, vol. 87, pp. 420–437, Oct. 2018, doi: 10.1016/j.future.2017.12.068.
- [13] J. Nocq, M. Celton, P. Gendron, S. Lemieux, and B. T. Wilhelm, "Harnessing virtual machines to simplify next-generation DNA sequencing analysis," *Bioinformatics*, vol. 29, no. 17, pp. 2075–2083, Sep. 2013, doi: 10.1093/bioinformatics/btt352.
- [14] P. Sheinidashtegol and M. Galloway, "Performance Impact of DDoS Attacks on Three Virtual Machine Hypervisors," in *2017 IEEE International Conference on Cloud Engineering (IC2E)*, Apr. 2017, pp. 204–214. doi: 10.1109/IC2E.2017.18.
- [15] S. Magomedov, D. Ilin, and E. Nikulchev, "Resource Analysis of the Log Files Storage Based on Simulation Models in a Virtual Environment," *Applied Sciences*, vol. 11, no. 11, Art. no. 11, Jan. 2021, doi: 10.3390/app11114718.
- [16] P. Zheng, Z. Zheng, X. Luo, X. Chen, and X. Liu, "A detailed and real-time performance monitoring framework for blockchain systems," in *Proceedings of the 40th International Conference on Software Engineering: Software Engineering in Practice*, New York, NY, USA, May 2018, pp. 134–143. doi: 10.1145/3183519.3183546.
- [17] M. M. Hassan and A. Rahman, "As Code Testing: Characterizing Test Quality in Open Source Ansible Development," in *2022 IEEE Conference on Software Testing, Verification and Validation (ICST)*, Apr. 2022, pp. 208–219. doi: 10.1109/ICST53961.2022.00031.
- [18] Z. Constantinescu and M. Vladoiu, "viztop – Intuitive Visualization of Remote Real-Time Monitoring of Linux Processes," in *2021 20th RoEduNet Conference: Networking in Education and Research (RoEduNet)*, Nov. 2021, pp. 1–6. doi: 10.1109/RoEduNet54112.2021.9638285.
- [19] E. Androulaki *et al.*, "Hyperledger fabric: a distributed operating system for permissioned blockchains," in *Proceedings of the Thirteenth EuroSys Conference*, New York, NY, USA, Apr. 2018, pp. 1–15. doi: 10.1145/3190508.3190538.
- [20] H. Lee *et al.*, "Multi-Batch Scheduling for Improving Performance of Hyperledger Fabric Based IoT Applications," in *2019 IEEE Global Communications Conference (GLOBECOM)*, Dec. 2019, pp. 1–6. doi: 10.1109/GLOBECOM38437.2019.9013551.
- [21] Д. Ю. Ильин, "Программный фреймворк для экспериментальной оценки характеристик информационно-технологических решений в виртуальной среде," *Russian Technological Journal*, vol. 10, no. 5, Art. no. 5, Oct. 2022, doi: 10.32362/2500-316X-2022-10-5-16-27.

# Choosing a technology stack for computing infrastructure for experimental research of digital currencies

A.S. Albychev, D.Y. Ilin

**Abstract**—One of the research areas of central bank digital currencies is an experimental assessment of characteristics of the underlying technologies. Commonly, blockchain technologies are considered for the implementation of digital currencies. The assessment of technical aspects requires an experimental setup which can be implemented using virtual machines. Due to the decentralized nature of digital currency technologies, the number of virtual machines can be quite large, therefore, to prepare the experimental setup, it is necessary to choose a stack of computing infrastructure technologies that is adequate for the task. The paper considers 2 infrastructure options for hosting an experimental setup of 33 software-controlled virtual machines. The virtual machines are provisioned with Hyperledger Fabric, resource monitor atop and other software supporting the functioning of the experimental setup and the blockchain-system. The experimental launch of the given setup was conducted using 2 technology stacks for computing infrastructure. Technology stack based on Windows Server and VirtualBox 7.1 did not allow the launch of the experimental setup on the presented hardware due to excessive and unproportional CPU resource usage. On the other hand, technology stack based on VMWare ESXi 7.0 hypervisor allowed the successful launch of the experimental setup. However, it requires configuration of service virtual machines for the functioning of the computing infrastructure.

**Keywords**—Central bank digital currency, blockchain, computing infrastructure, experimental setup.

## REFERENCES

- [1] M. N. Alsalmi, S. Ullah, and M. Rafique, "Accounting for Digital Currencies," *Research in International Business and Finance*, p. 101897, Feb. 2023, doi: 10.1016/j.ribaf.2023.101897.
- [2] N. Dashkevich, S. Counsell, and G. Destefanis, "Blockchain Application for Central Banks: A Systematic Mapping Study," *IEEE Access*, vol. 8, pp. 139918–139952, 2020.
- [3] T. Keister and D. Sanches, "Should Central Banks Issue Digital Currency?," *The Review of Economic Studies*, vol. 90, no. 1, pp. 404–431, Jan. 2023, doi: 10.1093/restud/rdac017.
- [4] I. Agur, A. Ari, and G. Dell'Ariccica, "Designing central bank digital currencies," *Journal of Monetary Economics*, vol. 125, pp. 62–79, Jan. 2022, doi: 10.1016/j.jmoneco.2021.05.002.
- [5] Y. Lee, B. Son, S. Park, J. Lee, and H. Jang, "A Survey on Security and Privacy in Blockchain-based Central Bank Digital Currencies," *Journal of Internet Services and Information Security*, vol. 11, no. 3, pp. 16–29, Aug. 2021, doi: 10.22667/JISIS.2021.08.31.016.
- [6] Y. Chu, J. Lee, S. Kim, H. Kim, Y. Yoon, and H. Chung, "Review of Offline Payment Function of CBDC Considering Security Requirements," *Applied Sciences*, vol. 12, no. 9, Art. no. 9, Jan. 2022, doi: 10.3390/app12094488.
- [7] T. Zhang and Z. Huang, "Blockchain and central bank digital currency," *ICT Express*, vol. 8, no. 2, pp. 264–270, Jun. 2022.
- [8] E. A. Opare and K. Kim, "A Compendium of Practices for Central Bank Digital Currencies for Multinational Financial Infrastructures," *IEEE Access*, vol. 8, pp. 110810–110847, 2020.
- [9] H. Xiong, M. Chen, C. Wu, Y. Zhao, and W. Yi, "Research on Progress of Blockchain Consensus Algorithm: A Review on Recent Progress of Blockchain Consensus Algorithms," *Future Internet*, vol. 14, no. 2, Art. no. 2, Feb. 2022, doi: 10.3390/fi14020047.
- [10] H. Javaid, C. Hu, and G. Brebner, "Optimizing Validation Phase of Hyperledger Fabric," in *2019 IEEE 27th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Oct. 2019, pp. 269–275.
- [11] S. Y. Jin and Y. Xia, "CEV Framework: A Central Bank Digital Currency Evaluation and Verification Framework With a Focus on Consensus Algorithms and Operating Architectures," *IEEE Access*, vol. 10, pp. 63698–63714, 2022.
- [12] J. Enes, R. R. Expósito, and J. Touriño, "BDWatchdog: Real-time monitoring and profiling of Big Data applications and frameworks," *Future Generation Computer Systems*, vol. 87, pp. 420–437, 2018.
- [13] J. Nocq, M. Celton, P. Gendron, S. Lemieux, and B. T. Wilhelm, "Harnessing virtual machines to simplify next-generation DNA sequencing analysis," *Bioinformatics*, vol. 29, no. 17, pp. 2075–2083, 2013.
- [14] P. Sheinidashtegol and M. Galloway, "Performance Impact of DDoS Attacks on Three Virtual Machine Hypervisors," in *2017 IEEE International Conference on Cloud Engineering (IC2E)*, Apr. 2017, pp. 204–214. doi: 10.1109/IC2E.2017.18.
- [15] S. Magomedov, D. Ilin, and E. Nikulchev, "Resource Analysis of the Log Files Storage Based on Simulation Models in a Virtual Environment," *Applied Sciences*, vol. 11, no. 11, Art. no. 11, 2021.
- [16] P. Zheng, Z. Zheng, X. Luo, X. Chen, and X. Liu, "A detailed and real-time performance monitoring framework for blockchain systems," in *Proceedings of the 40th International Conference on Software Engineering: Software Engineering in Practice*, New York, NY, USA, May 2018, pp. 134–143. doi: 10.1145/3183519.3183546.
- [17] M. M. Hassan and A. Rahman, "As Code Testing: Characterizing Test Quality in Open Source Ansible Development," in *2022 IEEE Conference on Software Testing, Verification and Validation (ICST)*, Apr. 2022, pp. 208–219. doi: 10.1109/ICST53961.2022.00031.
- [18] Z. Constantinescu and M. Vladioiu, "viztop – Intuitive Visualization of Remote Real-Time Monitoring of Linux Processes," in *2021 20th RoEduNet Conference: Networking in Education and Research (RoEduNet)*, Nov. 2021, pp. 1–6..
- [19] E. Androulaki *et al.*, "Hyperledger fabric: a distributed operating system for permissioned blockchains," in *Proceedings of the Thirteenth EuroSys Conference*, NY, USA, Apr. 2018, pp. 1–15.
- [20] H. Lee *et al.*, "Multi-Batch Scheduling for Improving Performance of Hyperledger Fabric Based IoT Applications," in *2019 IEEE Global Communications Conference (GLOBECOM)*, Dec. 2019, pp. 1–6.
- [21] D. Ilin, "Framework for experimental evaluation of software solutions in a virtual environment," *Russian Technological Journal*, vol. 10, no. 5, Art. no. 5, 2022..

## About authors

**Albychev Alexander Sergeevich**, deputy head of the Federal Treasury, The Ministry of Finance of the Russian Federation; head of the State financial technologies department, MIREA – Russian technological university

**Ilin Dmitry Yurievich**, Ph.D. (Tech.), associate professor at the department of Data processing digital technologies, MIREA – Russian technological university