

Суперкомпьютерные дни в России-2022,
Москва, 26-27 сентября 2022

Ensuring data integrity using Merkle trees in desktop grid systems

Обеспечение целостности данных в грид-системах
с помощью деревьев Меркла

И.И. Курочкин¹, С.В. Краснов²

¹ИППИ РАН, Москва,

²НИУ ВШЭ, Москва.

Российские суперкомпьютеры в top500*

Место в top500	Суперкомпьютер	Организация
22	Червоненкис	Яндекс
40	Галушкин	Яндекс
43	Ляпунов	Яндекс
46	Christofari Neo	Сбер
80	Christofari	Сбер
262	Ломоносов-2	МГУ
318	GROM	МТС

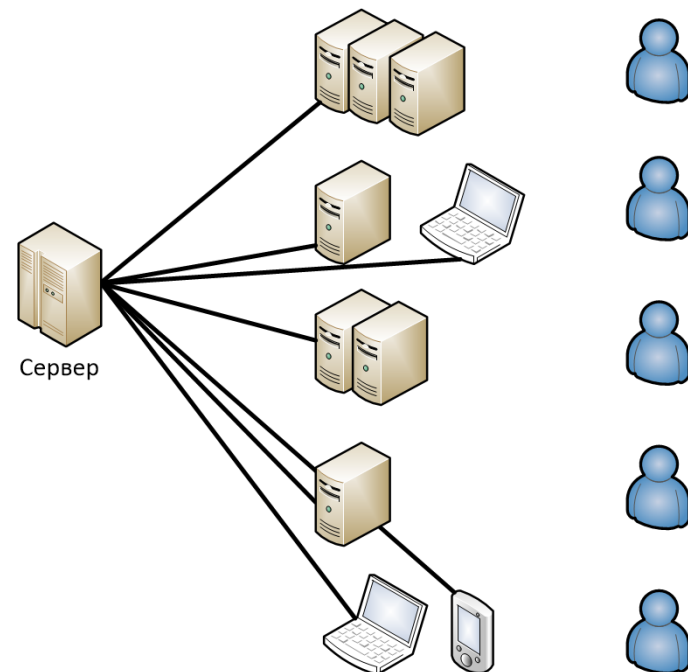
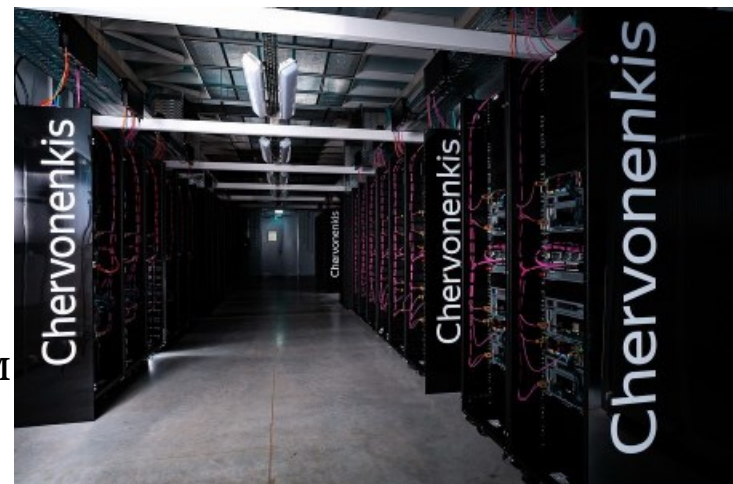
* Рейтинг 500 самых мощных суперкомпьютеров на июнь 2022 года

HPC and HTC

Для решения сложных вычислительных задач часто используют многопроцессорные вычислительные системы (МВС).

Класс МВС достаточно широк и к нему можно отнести достаточно много типов вычислительных систем, в том числе системы с сильной связью между вычислителями (SMP-системы) и системы со слабой связью: вычислительные кластеры и системы с архитектурой MPP. Использование высокопроизводительных вычислений (**high performance computing, HPC**) позволяет быстро решать сложные вычислительные задачи, но сопряжено с высокими финансовыми затратами на содержание больших МВС.

При отсутствии возможности использования HPC для решения ряда задач можно использовать другой подход (**high throuput computing, HTC**) и распределенные вычислительные системы.



Грид-системы из персональных компьютеров

Развитие ГСПК стало возможным из-за появления у рядового пользователя достаточно мощных вычислительных устройств и увеличения доступности к глобальной сети Интернет.

Одной из основных причин появления ГСПК является использование простаивающих вычислительных ресурсов, так у большинства пользователей персональных компьютеров уровень использования находится на уровне 10-15%.

Одним из подвидов грид-систем из персональных компьютеров (ГСПК) являются публичные ГСПК, в которых вычислительные мощности предоставляются добровольцами.

Такие публичные ГСПК называют проектами добровольных распределенных вычислений (ДРВ) и часто дают им название по научной проблеме или вычислительной задаче, которая решается в рамках проекта, к примеру, SiDock@home, Gerasim@home, LHC@home, Asteroids@home.

Размер таких ГСПК может быть большим и достигать миллионов вычислительных узлов, как в проекте Folding@home.



Платформы для организации распределенных вычислений

- HTCondor
- Globus
- BOINC
- Oracle Grid Engine



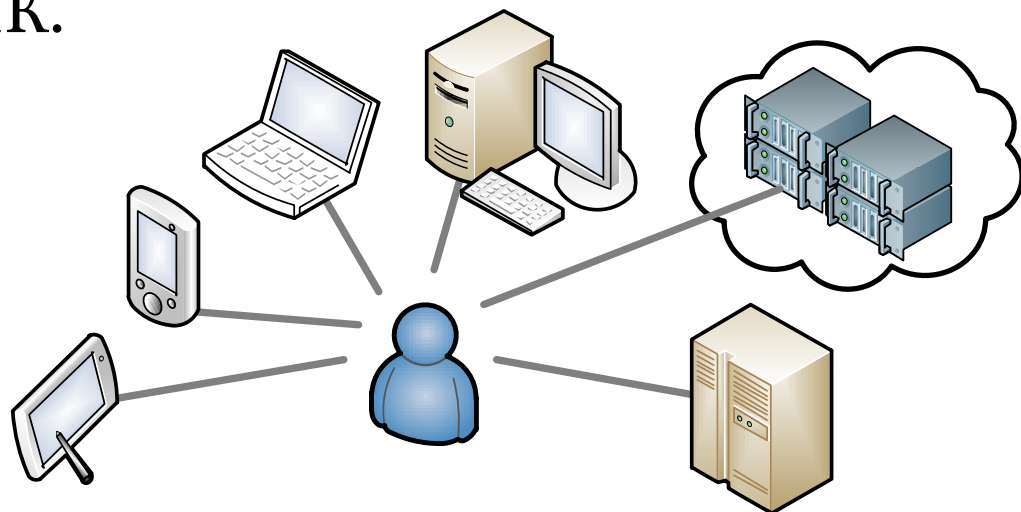


Платформа BOINC

BOINC – Berkeley Open Infrastructure for Network Computing

Платформа для организации добровольных распределенных вычислений:

- состоит из серверной и клиентской части;
- дает возможность задействовать вычислительные мощности персональных компьютеров(ПК);
- кроссплатформенная клиентская часть;
- гибкая настройка клиентской части для эффективного использования ресурсов ПК.



Кто такие добровольцы?

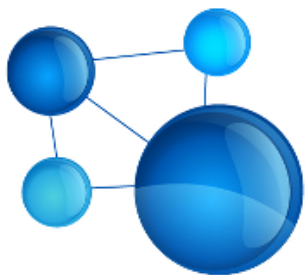
Доброволец (cruncher) предоставляет свои вычислительные ресурсы для расчета научных распределенных проектов

Почему?

- ✓ Желание помочь науке
- ✓ Причастность к научным открытиям (желание разобраться в получаемых результатах)
- ✓ **Спортивный интерес** (кто больше наберет баллов)
- ✓ Общение
- ✓ Участие в социально значимых проектах

Спортивная составляющая и неспортивное поведение

- Баллы
- Рейтинг
- Саботаж проекта
- Криптовалюты



Curecoin (CURE)



Gridcoin

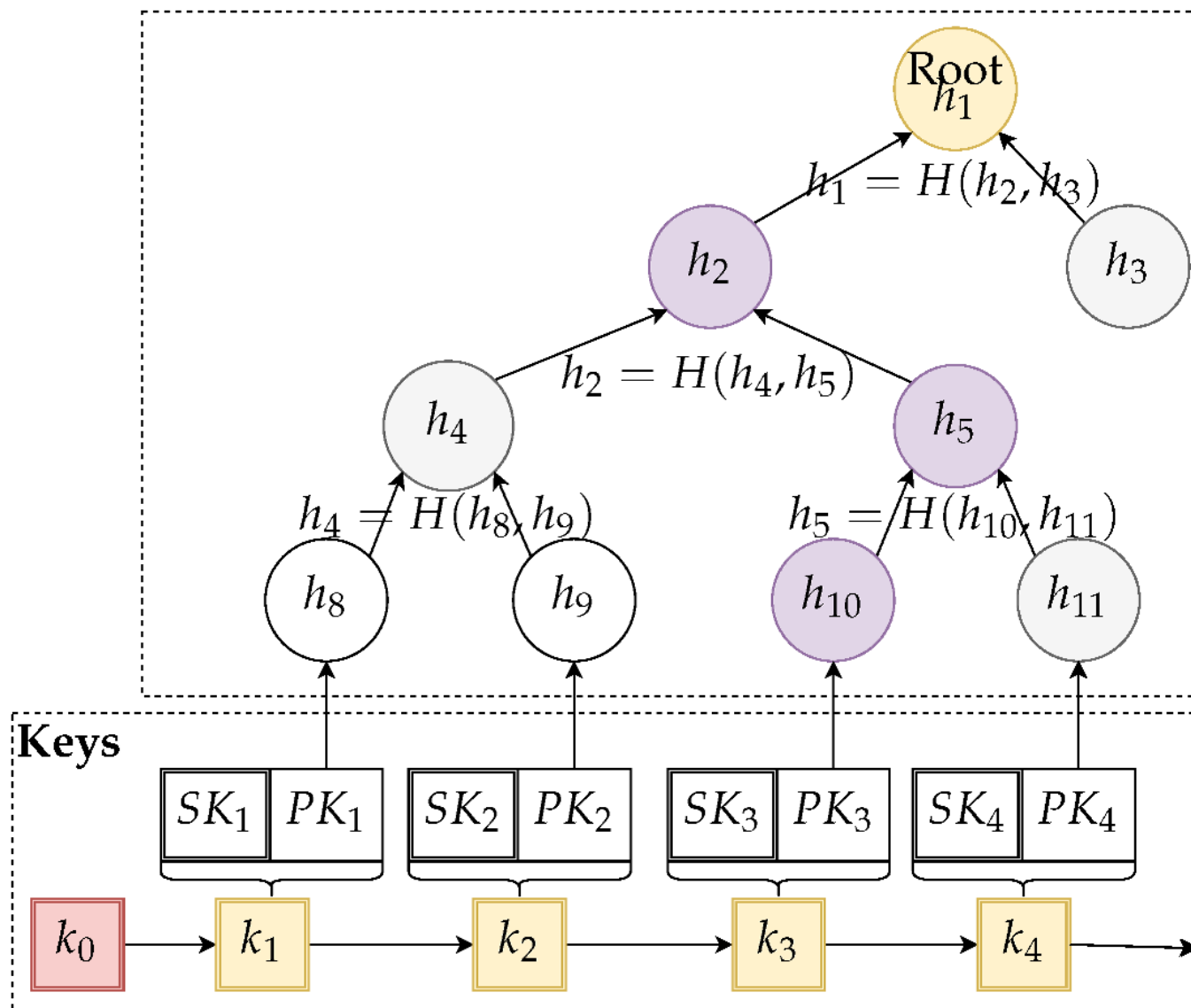
Обмен данными



Угрозы обеспечения целостности данных

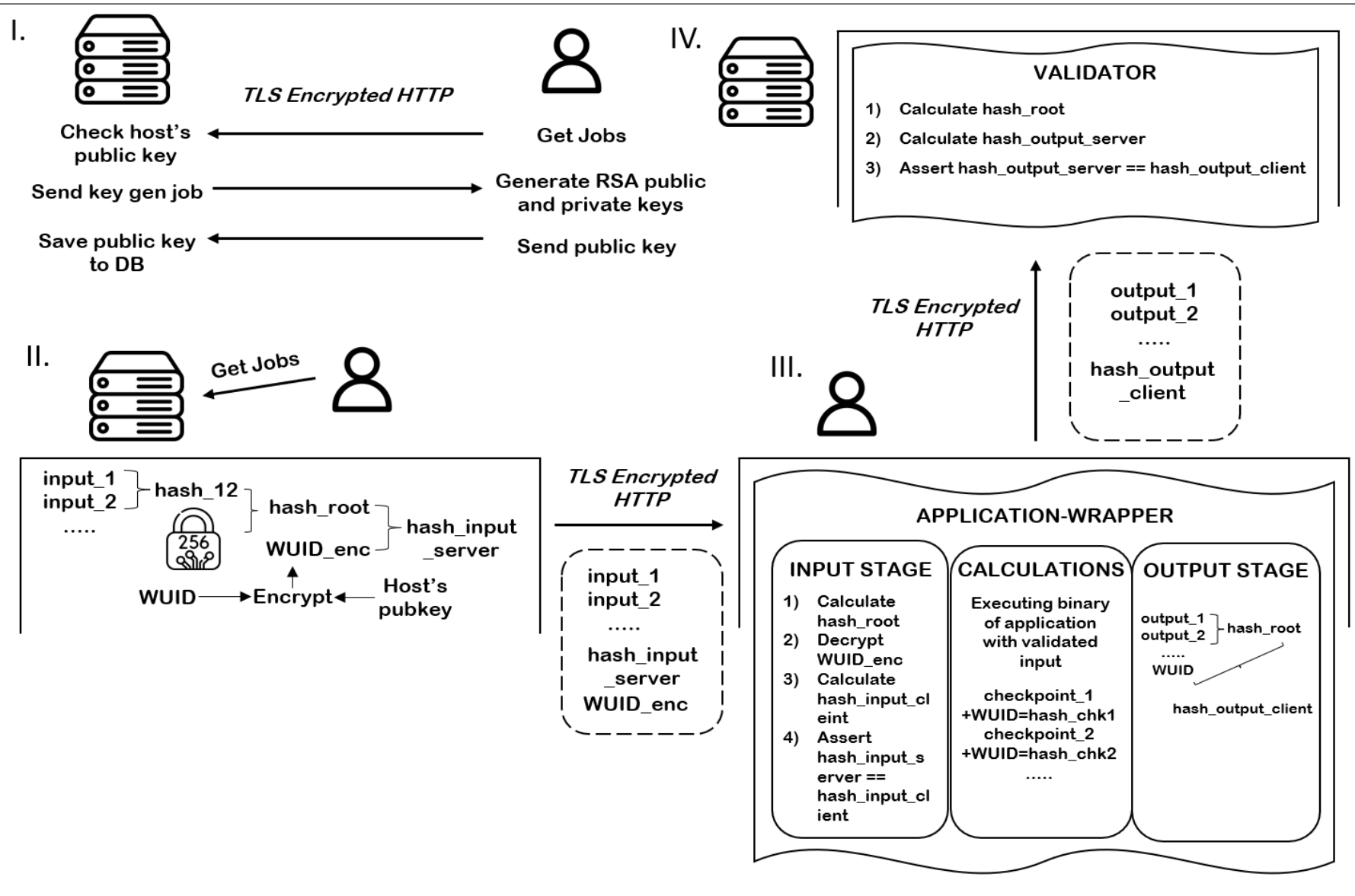
- Модификация/ошибки сетевого трафика
- Модификация/изменение входных файлов
- Модификация/изменение чекпоинтов приложения
- Модификация/изменение выходных файлов

Дерево Меркла



ethereum

Предложенный подход



Особенности

- Передача данных по зашифрованному каналу TLS/SSL
- Добавление workunit ID или WUID (генерация на стороне сервера)
- Алгоритм хеширования SHA-256 вместо MD-5
- Возможность работы в 2 режимах с вычислительным приложением
 - Исполняемый файл + обертка
 - Модифицированное приложение

Тестирование

- Для приложения **KLRMD*** производительность измерялась на 36 расчетных заданиях (36 workunits), с разным набором входных параметров приложения.
- Вычисления производились на персональном компьютере с 6-ядерным (12 потоков) процессором Intel Core i5 10600K 4100 GHz и 32 GB ОЗУ на ОС Windows 11.
- Для набора заданий, которые вычислялись без механизма целостности данных, параметр репликации был установлен в 2 (*min_quorum*), что эквивалентно 2 вычислительным результатам для 1-го задания (workunit'a).
- Для набора заданий с обеспечением целостности данных параметр репликации был снижен в два раза и составлял 1.

* Задача нахождения канонических форм (КФ) ДЛК, одна из задач анализа различных характеристик ортогональных диагональных латинских квадратов

Сравнение объемов данных

Без механизма обеспечения целостности данных				С механизмом обеспечения целостности данных			
Входные файлы	Размер, байт	Выходные файлы	Размер, байт	Входные файлы	Размер, байт	Выходные файлы	Размер, байт
in	568	stdout	37 728	in	568	stdout	37 728
in_dummy_1	345	stderr.txt	19 616	in_dummy_1	345	stderr.txt	19 616
in_dummy_2	652	out	1	in_dummy_2	652	out	1
in_dummy_3	126			in_dummy_3	126	hash_output_client	64
in_dummy_4	432			in_dummy_4	432	input_validator.log	247
				hash_input_server	64	output_submitter.log	542
				WUID_enc	350		
Сумма, байт	2 123		57 345	Сумма, байт	2 537		58 198
Общий объем данных, байт		59 468		Общий объем данных, байт		60 735	
		Увеличение данных			2,13%		

Моделирование угроз

```
...
<task>
  <application>output_submitter.exe</application>
  <command_line>--output stdout out</command_line>
</task>
<task>
  <application>sleep.exe 10</application>
</task>
</job_desc>
```

Если валидация нарушена, то в логах output_validator:

```
# output_validator.log:
```

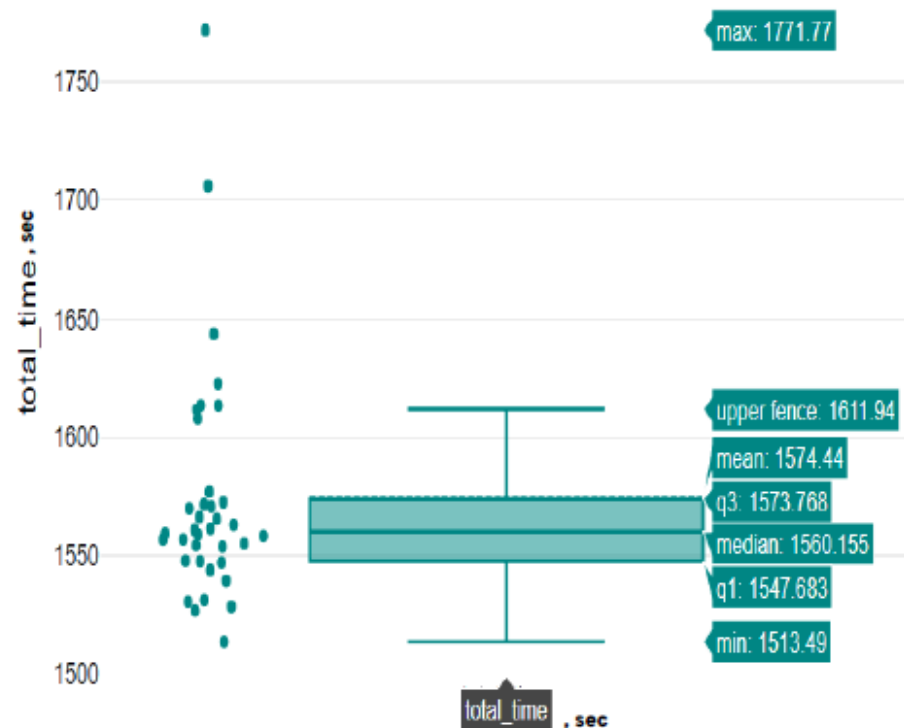
```
14:21:34,173 root ERROR Integrity output validation has failed: integrity hashes are
not equal for output files
```

Если валидация прошла успешно:

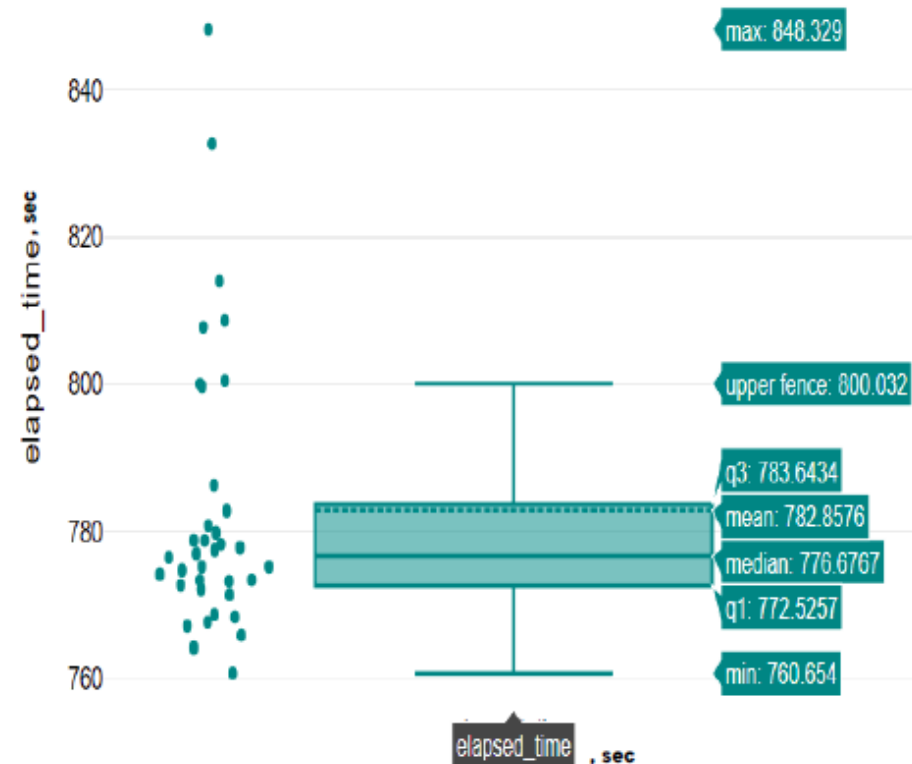
```
12:15:18,990 root INFO Validation has successfully passed: server integrity hash
603029baa1e93a3483cf39bff971c06c6dadafd4aa337943219c7c202bd103d2
equals to client integrity hash
603029baa1e93a3483cf39bff971c06c6dadafd4aa337943219c7c202bd103d2
# output_hash_server == output_hash_client
```


Результаты эксперимента

Без обеспечения целостности данных



С обеспечением целостности данных



Заключение

- ✓ Были определены наиболее распространенные угрозы для целостности данных;
- ✓ Предложен подход обеспечения целостности данных на основе деревьев Меркла;
- ✓ Внедрение предложенного подхода в проект на платформе BOINC позволило уменьшить влияние атак на ход вычислительного эксперимента;
- ✓ Тестирование предложенного подхода показало, что производительность в рамках одного вычислительного результата, а также объем данных практически не изменились
- ✓ При уменьшении значения параметра репликации удалось повысить производительность до значения, близкого к верхней оценке.



Спасибо за внимание

Github-репозиторий с исходным кодом компонент целостности данных:

https://github.com/jellythefish/boinc_helpers

Github-репозиторий с клоном исходных кодов BOINC с модификациями сервера:

https://github.com/jellythefish/boinc_improved_fault_tolerance_and_integrity/pull/1

Центр распределенных вычислений ИППИ РАН

web: distributed-computing.ru

e-mail: qurochkin@gmail.com, kurochkin@iitp.ru