

Работа с серверами и протокол удалённого управления SSH

Владислав Шевченко

Ведущий инженер разработки и внедрения моделей машинного обучения Альфа-Банке

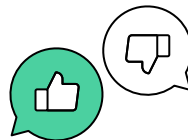


Проверка связи





Если у вас нет звука:

- убедитесь, что на вашем устройстве и на колонках включён звук
- обновите страницу вебинара (или закройте страницу и заново присоединитесь к вебинару)
- откройте вебинар в другом браузере
- перезагрузите компьютер (ноутбук) и заново попытайтесь зайти



Поставьте в чат:

-  если меня видно и слышно
-  если нет

Владислав Шевченко

О спикере:

- ментор программы «Инженерия данных»
- ведущий инженер разработки и внедрения моделей машинного обучения в Альфа-Банке
- преподаватель дисциплины «Семинар наставника»



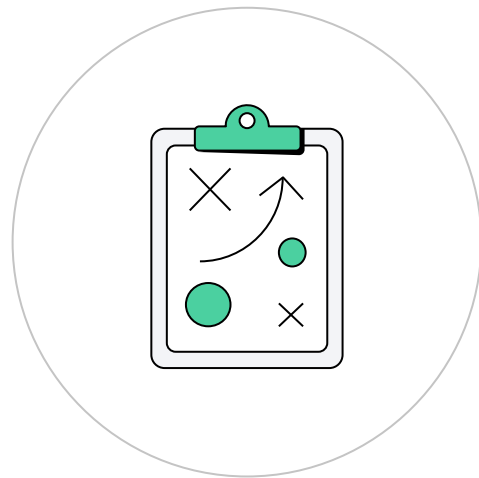
Правила участия

- 1 Приготовьте блокнот и ручку, чтобы записывать важные мысли и идеи
- 2 Продолжительность воркшопа — 80 минут
- 3 Вы можете писать свои вопросы в чате
- 4 Запись воркшопа будет доступна в LMS
- 5 Обсуждения можно продолжить в Telegram
- 6 Требуемая операционная система — Unix



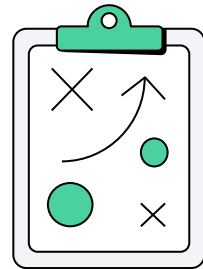
Цели занятия

- 1 Разберём практические задачи по работе с SSH
- 2 Изучим продвинутые методы решения задач



План занятия

- 1 Практика. Решение задач по работе SSH
- 2 Практика. Решение задач на основе продвинутых методов



Практика

Решение задач по работе SSH



1

Упражнение 1

Резервное копирование файлов и перенос на другой сервер

Задание:

Напишите **Bash**-скрипт, который создаёт резервную копию указанной директории, архивирует её, а затем копирует архив на другой сервер по **SSH**.

Требования:

- Создание архивов указанной директории с помощью **tar**
- Использование **scp** для копирования архива на удалённый сервер
- Автоматическое удаление старых архивов на удалённом сервере (например, если их больше трёх).

Тайминг: 15 минут

Упражнение 2

Автоматическое обновление приложений на удалённом сервере

Задание:

Напишите скрипт, который подключается к удалённому серверу по **SSH**, проверяет наличие обновлений для системы или приложений, устанавливает обновления и перезагружает сервер, если это необходимо.

Требования:

- Подключение к серверу по **SSH**
- Выполнение команд обновления системы
- Отправка уведомления по электронной почте, если сервер был перезагружен

Тайминг: 15 минут

Упражнение 3

Архивирование файлов и скачивание с сервера с последующим разархивированием

Задание:

Напишите скрипт, который подключается к удалённому серверу, архивирует указанную директорию, скачивает архив на локальную машину и разархивирует его.

Требования:

- Архивирование удалённой директории
- Скачивание архива на локальный компьютер с помощью scp
- Разархивирование скачанного архива

Тайминг: 15 минут

Упражнение 4

Мониторинг свободного места на сервере и отправка уведомлений

Задание:

Создайте скрипт, который подключается к удалённому серверу и проверяет свободное место на диске. Если свободное место меньше заданного порога, скрипт отправляет уведомление по электронной почте.

Требования:

- Удалённая проверка свободного места на диске
- Сравнение с заданным порогом
- Отправка уведомления по электронной почте, если место заканчивается

Тайминг: 15 минут



Ваши вопросы?



Практика

Решение задач на основе продвинутых методов



2

Упражнение 5

Синхронизация файлов между локальной машиной и удалённым сервером

Задание:

Напишите скрипт, который синхронизирует файлы между локальной директорией и удалённой с использованием **rsync** по **SSH**.

Требования:

- Синхронизация данных в обоих направлениях
- Игнорирование определённых типов файлов или директорий
- Отправка отчёта по электронной почте о завершении синхронизации

Тайминг: 15 минут

Упражнение 6

Автоматический запуск команды на нескольких серверах

Задание:

Создайте скрипт, который выполняет указанную команду на нескольких серверах, подключаясь по SSH.

Требования:

- Перечень серверов должен быть в текстовом файле
- Скрипт должен подключаться к каждому серверу и выполнять заданную команду
- Вывод команд сохраняется в локальный лог-файл

Тайминг: 15 минут

Упражнение 7

Мониторинг загрузки процессора и автоматическое завершение процессов

Задание:

Создайте скрипт, который подключается к удалённому серверу, проверяет загрузку процессора и завершает процессы, если нагрузка превышает заданный порог.

Требования:

- Удалённая проверка текущей загрузки процессора (uptime или top)
- Сравнение средней загрузки с установленным порогом
- Автоматическое завершение выбранных процессов, если нагрузка выше порога

Тайминг: 15 минут



Ваши вопросы?

Итоги занятия

- Разобрали практические задачи по работе с SSH
- Изучили продвинутые методы решения задач



Рефлексия

- Что изменилось? «Раньше я думал(а), что..., а теперь...»
- Какие вопросы у меня остались?



Следующий вебинар

→ Работа с Dokcer



Работа с серверами и протокол удалённого управления SSH

Владислав Шевченко

Ведущий инженер разработки и внедрения моделей машинного обучения Альфа-Банке

