

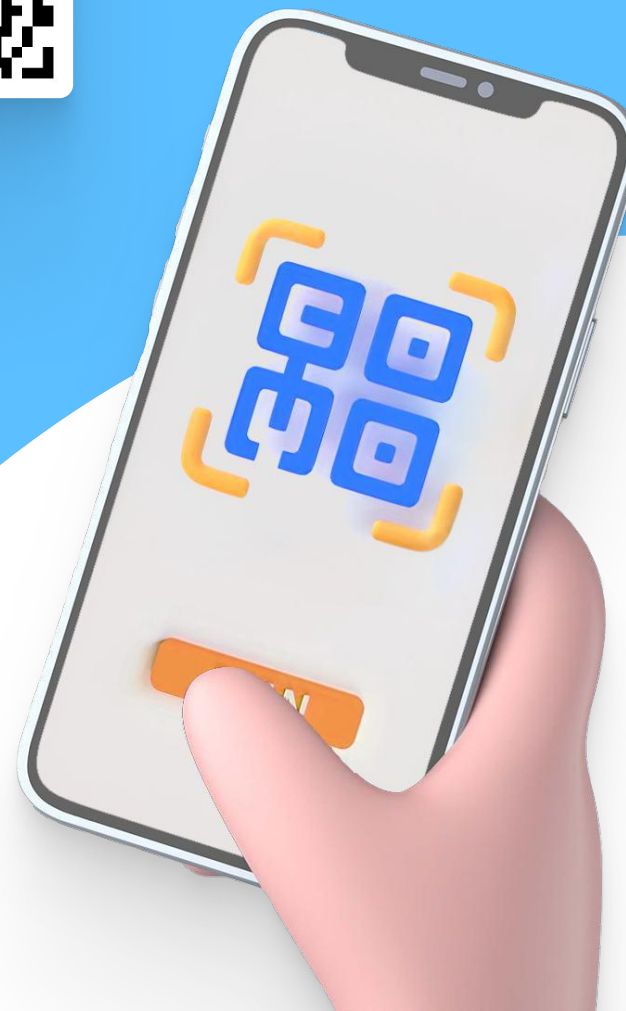


КУПЛЕНО НА
SKLADCHIK.ORG

Все самые свежие
курсы по лучшей цене!



*Отсканируйте
QR-код телефоном*



Linux. Уровень 1. Основы администрирования

Русанов Сергей





О преподавателе

Русанов Сергей Александрович



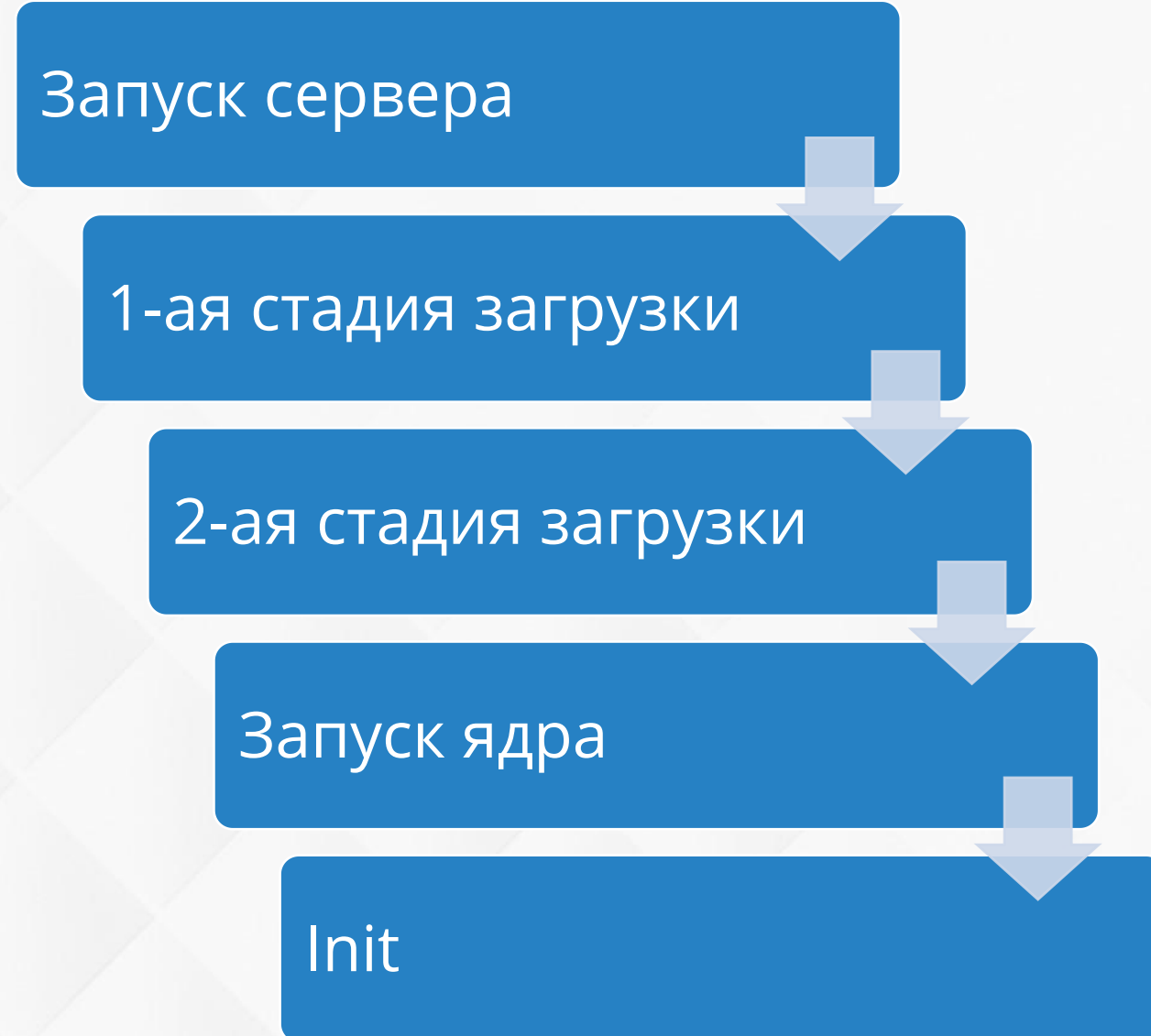
- ✓ Linux
- ✓ Ubuntu
- ✓ Centos
- ✓ Nginx
- ✓ Apache
- ✓ Docker

Модуль 8. Системы инициализации и управление сервисами





Запуск GNU/Linux



BIOS. Выбор загрузочного носителя

Загрузочная запись GPT

Загрузчик GRUB

Linux Kernel. Запуск Init

Запуск всех остальных процессов - инициализация системы



Обзор систем инициализации

- **SysV** (System V init) - одна из первых систем инициализации в Linux. Она базируется на скриптах и использует директорию `/etc/init.d/` для хранения и запуска скриптов инициализации. Последовательный запуск служб.
- **OpenRC** - альтернативная система инициализации, используемая в некоторых дистрибутивах Linux, таких как Gentoo. OpenRC основан на простых скриптах и позволяет гибко управлять сервисами и задачами при загрузке системы. Поддерживает параллельный запуск служб.
- **systemd** - одна из самых популярных и современных систем инициализации в современных дистрибутивах Linux от Fedora. Systemd использует параллельную инициализацию для ускорения загрузки, предоставляет расширенные возможности для управления службами, событиями и журналированием. Параллельный запуск служб. Не совместим с SysV.
- **Upstart** - это гибридная система инициализации, она использует как SysV скрипты запуска, так и файлы служб Systemd. Разработана для Ubuntu.
- **runit** - кроссплатформенная, легковесная и эффективная система инициализации, которая проста в настройке и обеспечивает надежную и стабильную работу системы. Портативность.



Systemd типы Unit

- **target** — группирует модули
- **service** — отвечает за запуск сервисов (служб) и поддерживает вызов интерпретаторов для исполнения пользовательских скриптов
- **mount** — занимается монтированием файловых систем
- **automount** — автоматическое монтирование файловых систем, используется при обращении к точке монтирования;
- **swap** — отвечает за подключение файла подкачки
- **timer** — запускает модули по расписанию, аналог cron
- **socket** — запуск модуля при подключении к сокету
- **slice** — группировка других модулей в контейнер (дерево) cgroups
- **device** — использует реакцию на подключение какого-либо устройства
- **path** — запуск модуля по событию доступа по конкретному пути в файловой системе



systemd

```
/usr/lib/systemd/system/  
/run/systemd/system/  
/etc/systemd/system/
```

```
[Unit]
```

```
Description=NMON monitoring  
Requires=network.target  
After=network.target  
Before=nginx.target  
Wants=apache.target
```

```
[Service]
```

```
Type=notify  
ExecStart=/usr/bin/nmon -V  
ExecReload=/bin/kill -HUP $MAINPID  
KillMode=process  
Restart=on-failure
```

```
[Install]
```

```
WantedBy=multi-user.target
```

```
#юниты из пакетов
```

```
#юниты, сгенерированные автоматически
```

```
#пользовательские юниты
```

```
#описание
```

```
#зависимости. Запуск только при запущенном
```

```
#зависимости. Запуск сервиса ПОСЛЕ network
```

```
#зависимости. Запуск сервиса ДО nginx
```

```
#зависимости. Для запуска желателен
```

```
#тип юнита
```

```
#тип сервиса
```

```
#команда запуска
```

```
#команда перезапуска
```

```
#команда завершения или как завершать
```

```
#перезапуск в случае отказа
```

```
#тип запуска, многопользовательский
```



systemd типы service

- **simple** (по умолчанию) – происходит незамедлительный запуск этой службы, с учетом того что процесс не разветвляется (fork). Не используйте simple если пользуетесь очередностью запуска. Одно исключение это активация сокета
- **forking** – служба считается запущенной после того, после разветвления процесса с завершением родительского процесса. Используется для запуска классических демонов исключая случаи, когда в таком поведении процесса нет необходимости. Также желательно указать PIDFile=, чтобы systemd мог отслеживать основной процесс.
- **oneshot** – удобен для скриптов, которые выполняют одно задание и завершаются. При необходимости можно задать параметр RemainAfterExit=yes, чтобы systemd считал процесс активным даже после его завершения
- **notify** – идентичен параметру simple, но с оговоркой, что демон пошлет systemd сигнал о своей готовности. Эталонная реализация данного уведомления представлена в libsystemd-daemon.so.
- **dbus** – служба считается находящейся в состоянии готовности, когда указанный параметр BusName появляется в системной шине Dbus
- **idle** – откладывается выполнение двоичного файла службы до момента выполнения всех остальных задач. В остальном поведение аналогично simple



systemctl

```
systemctl status <имя_сервиса>
```

#статус

```
systemctl enable <имя_сервиса>
```

#автозапуск

```
systemctl disable <имя_сервиса>
```

#отключение автозапуска

```
systemctl start <имя_сервиса>
```

#запуск

```
systemctl stop <имя_сервиса>
```

#остановка

```
systemctl restart <имя_сервиса>
```

#перезапуск

```
systemctl list-units
```

#СПИСОК АКТИВНЫХ ЮНИТОВ

```
systemctl list-units --all
```

#СПИСОК ВСЕХ ЮНИТОВ

```
systemctl daemon-reload
```

#применение обновленной конфигурации

```
systemctl cat <имя_сервиса>
```

#просмотр unit-файл сервиса

Лабораторная работа «systemd»



Модуль 9. Управление ПО





Обзор вариантов управления ПО

пакетный менеджер

- apt или apt-get для Debian, Ubuntu
- yum или dnf для Fedora, Centos, RedHat
- pacman для Arch
- zypper для openSUSE

установочные пакеты

- dpkg для Ubuntu, Debian
- rpm для Fedora, Centos, RedHat, AltLinux

сборка из исходников

- configure (месим тесты) - подготовка исходников
- make (в печь) - сборка
- make install (кушать подано!) - установка программы



Работа с ПО из репозиториев. Apt

```
/etc/apt/sources.list  
/etc/apt/sources.list.d/  
/etc/apt/trusted.gpg.d/
```

```
#список репозиториев  
#каталог для репозиториев  
#каталог ключей
```

Компоненты (секции) репозиториев (для Ubuntu):

Main – свободное ПО, официально поддерживаемое компанией Canonical.

Restricted – проприетарное ПО (в основном — драйверы устройств), официально поддерживаемое компанией Canonical.

Universe – свободное ПО, официально не поддерживаемое компанией Canonical (но поддерживаемое сообществом пользователей).

Multiverse – проприетарное ПО, не поддерживаемое компанией Canonical.

```
deb https://apt.postgresql.org/pub/repos/apt/ jammy-pgdg main
```

```
deb  
https://apt.postgresql.org/pub/repos/apt/  
jammy-pgdg  
main
```

```
#тип пакетов  
#адрес репозитория  
#версия ОС  
#branch (секция)
```

Пример стандартного файла с репозиториями для Ubuntu 22.04



```
# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.
deb http://ru.archive.ubuntu.com/ubuntu jammy main restricted
# deb-src http://ru.archive.ubuntu.com/ubuntu jammy main restricted

## Major bug fix updates produced after the final release of the
## distribution.
deb http://ru.archive.ubuntu.com/ubuntu jammy-updates main restricted
# deb-src http://ru.archive.ubuntu.com/ubuntu jammy-updates main restricted

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team. Also, please note that software in universe WILL NOT receive any
## review or updates from the Ubuntu security team.
deb http://ru.archive.ubuntu.com/ubuntu jammy universe
# deb-src http://ru.archive.ubuntu.com/ubuntu jammy universe
deb http://ru.archive.ubuntu.com/ubuntu jammy-updates universe
# deb-src http://ru.archive.ubuntu.com/ubuntu jammy-updates universe

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team, and may not be under a free licence. Please satisfy yourself as to
## your rights to use the software. Also, please note that software in
## multiverse WILL NOT receive any review or updates from the Ubuntu
## security team.
deb http://ru.archive.ubuntu.com/ubuntu jammy multiverse
# deb-src http://ru.archive.ubuntu.com/ubuntu jammy multiverse
deb http://ru.archive.ubuntu.com/ubuntu jammy-updates multiverse
# deb-src http://ru.archive.ubuntu.com/ubuntu jammy-updates multiverse

## N.B. software from this repository may not have been tested as
## extensively as that contained in the main release, although it includes
## newer versions of some applications which may provide useful features.
## Also, please note that software in backports WILL NOT receive any review
## or updates from the Ubuntu security team.
deb http://ru.archive.ubuntu.com/ubuntu jammy-backports main restricted universe multiverse
# deb-src http://ru.archive.ubuntu.com/ubuntu jammy-backports main restricted universe multiverse

deb http://ru.archive.ubuntu.com/ubuntu jammy-security main restricted
# deb-src http://ru.archive.ubuntu.com/ubuntu jammy-security main restricted
deb http://ru.archive.ubuntu.com/ubuntu jammy-security universe
# deb-src http://ru.archive.ubuntu.com/ubuntu jammy-security universe
deb http://ru.archive.ubuntu.com/ubuntu jammy-security multiverse
# deb-src http://ru.archive.ubuntu.com/ubuntu jammy-security multiverse
```

Принцип работы пакетного менеджера

КАРТИНКА. ИЛЛЮСТРИРУЮЩАЯ РАБОТУ ПАКЕТНОГО МЕНЕДЖЕРА



Работа с apt

```
man apt  
apt --help
```

```
apt update  
apt upgrade
```

```
apt install <пакет1>  
apt install <пакет1> <пакет2> <пакет3>  
apt remove <пакет1>  
apt autoremove
```

```
apt list --installed  
apt list --upgradable  
apt list --all-version
```

```
apt search <слово>  
apt search --names-only <имя_пакета>  
apt show <пакет1>
```

```
apt edit-sources
```

```
#справка apt  
#справка apt
```

```
#обновление списка пакетов  
#обновление установленных пакетов
```

```
#установка пакета «пакет1»  
#установка пакетов нескольких пакетов  
#удаление пакета «пакет1»  
#удаление неиспользуемых зависимостей
```

```
#список установленных пакетов  
#список пакетов, требующих обновления  
#список пакетов для вашей системы
```

```
#поиск пакета по названию и описанию  
#поиск пакета по названию  
#описание пакета «пакет1»
```

```
#редактирование файла с репозиториями
```



Лабораторная работа «Управление ПО»



Модуль 10 . Управление локальными пользователями и группами



Обзор конфигурационных файлов

<code>/etc/passwd</code>	<code>#информация о пользователях системы</code>
<code>/etc/group</code>	<code>#информация о группах пользователей</code>
<code>/etc/fstab</code>	<code>#точки монтирования файловых систем</code>
<code>/etc/hosts</code>	<code>#соответствие IP и имен хостов</code>
<code>/etc/resolv.conf</code>	<code>#настройки DNS</code>
<code>/etc/network/interfaces</code>	<code>#настройки сетевых интерфейсов (без Netplan)</code>
<code>/etc/sysctl.conf</code>	<code>#параметры ядра</code>
<code>/etc/ssh/sshd_config</code>	<code>#конфигурационный файл OpenSSH</code>
<code>/etc/sudoers</code>	<code>#параметры пользователей sudo</code>



/etc/passwd и /etc/group



```
ubuntuadmin:x:1000:1000:Ubuntu Admin:/home/ubuntuadmin:/bin/bash
```

```
ubuntuadmin
```

```
x
```

```
1000:1000
```

```
0
```

```
1-99
```

```
100-999
```

```
от 1000
```

```
Ubuntu Admin
```

```
/home/ubuntuadmin
```

```
/bin/bash
```

```
/usr/sbin/nologin
```

```
#имя пользователя (login)
```

```
#пароль зашифрован и хранится в /etc/shadow
```

```
#UID (User ID) и GID (Group ID)
```

```
#UID зарезервированный под root
```

```
#предустановленные пользователи
```

```
#системный пользователи
```

```
#обычные пользователи
```

```
#Комментарий к пользователю ФИО, должность и т.п.
```

```
#домашний каталог
```

```
#оболочка по умолчанию
```

```
#запрет на подключение к оболочке
```

```
testgroup:x:1002:test1,petrov,ivanov
```

```
testgroup
```

```
x
```

```
1002
```

```
test1,petrov,ivanov
```

```
#имя группы
```

```
#пароль зашифрован и хранится в /etc/shadow
```

```
#GID (Group ID)
```

```
#пользователи, входящие в данную группу
```

/etc/sudoers



```
ТИП имя_алиаса = значение1, значение2, значение3
ТИП                                     #тип алиаса (пользователи, группы или команды)
имя_алиаса                             #имя алиаса, котрое можно использовать
значение1                               #пользователь, группа или команда

User_Alias      ADMINS = petrov, ivanov, sidorov
User_Alias      SUDO_USERS = %sudo
Cmnd_Alias      PKGMAN = /usr/bin/apt, /usr/bin/dpkg

ПОЛЬЗОВАТЕЛЬ хост = (другой_пользователь:группа) команды
ПОЛЬЗОВАТЕЛЬ    #пользователь или группа, для кого правило
хост            #для какого узла применяется правило
другой_пользователь:группа #конкрутный пользователь и пользователи из группы
команды        #команда или набор команд

%sudo ALL = (ALL) ALL
%sudo ALL = (root:admins) ALL
ADMINS host1 NOPASSWD:PKGMAN
petrov ALL NOPASSWD:/usr/bin/mount, /usr/bin/umount
```

Управление пользователями

```
useradd petrov  
adduser ivanov  
passwd petrov  
usermod petrov  
pinky  
userdel petrov
```

```
#создание пользователя petrov  
#создание пользователя ivanov  
#смена пароля пользователю petrov  
#изменение пользователя  
#информация об активных сессиях и пользователях  
#удаление пользователя
```



Управление группами

```
groupadd group1  
groupmod group1  
usermod -a -G group1 test  
groupdel group1
```

```
#создание группы group1  
#изменение группы group1  
#добавление пользователя test в группу group1  
#удаление группы group1
```



Повышение привилегий



```
sudo <команда>  
su  
su -  
sudo su  
sudo su -
```

```
#Выполнение команды из под root с вводом пароля пользователя  
#суперпользователь в текущем каталоге (пароль root)  
#суперпользователь в каталоге /root (пароль root)  
#суперпользователь в текущем каталоге с паролем пользователя  
#суперпользователь в каталоге /root с паролем пользователя
```

Лабораторная работа «Управление пользователями»



Модуль 11 . Контроль доступа



Виды контроля доступа

DAC (Дискреционный контроль доступа)

- Владелец:Группа:Другие_пользователи

MAC (Мандатный контроль доступа)

- доступ на основе политик безопасности

RBAC (Контроль доступа на основе ролей)

- Разрешения -> Роль -> Пользователи

ACL (Списки контроля доступа)

- DAC+

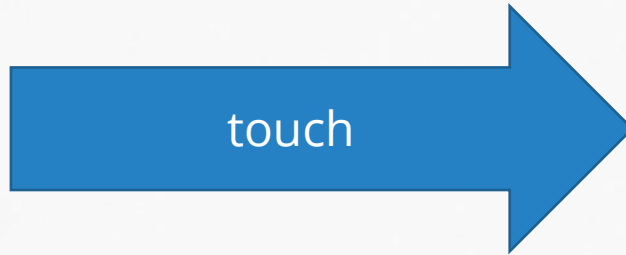
Владение



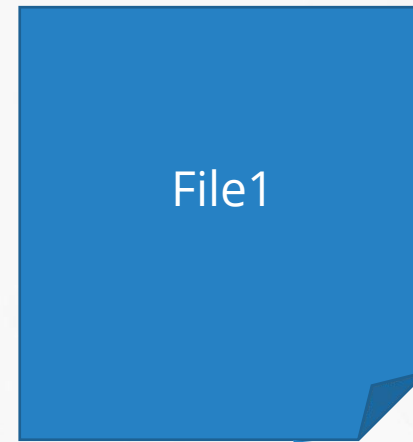
Petrov

Первичная группа -
group1

Дополнительная группа
- group2



touch



File1



Petrov

Владелец

Первичная группа -
group1

группа владельцев

Владение. Команда chown



ТОЛЬКО владелец может изменить владельца (ну еще root)

chown (change owner) <КТО?> <ЧТО>

#общий синтаксис

chown petrov /opt/data/file1

#пользователь petrov - новый владелец file1

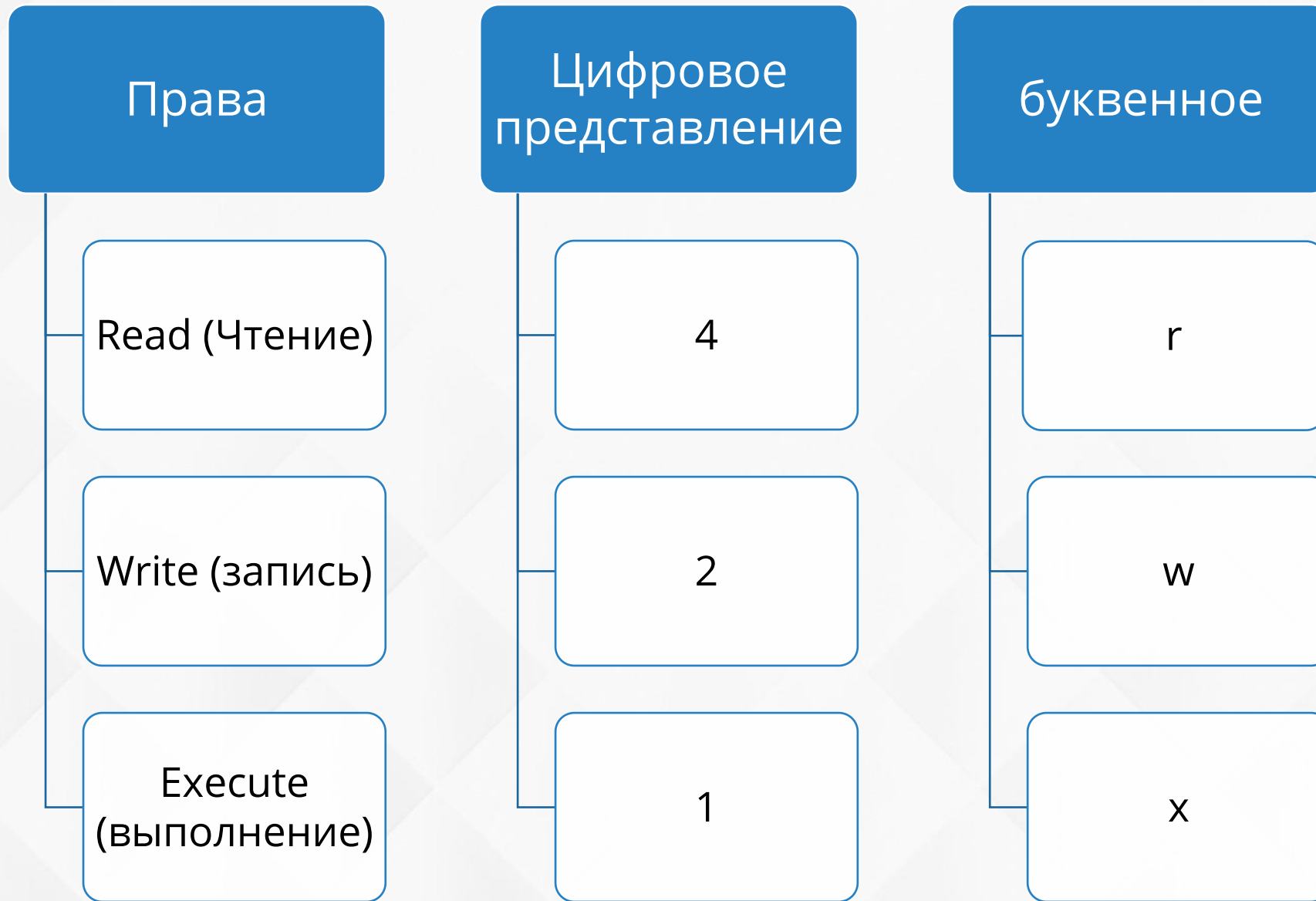
```
drwxrwxr-x 3 user user 4096 Mar 29 09:50 dir1
drwxrwxr-x 2 user user 4096 Mar 29 06:47 dir2
drwxrwxr-x 2 user user 4096 Mar 29 06:48 dir3
-rw-rw-r-- 1 user group1 1351 Mar 25 19:30 ls.txt
-rw-rw-r-- 1 root user 0 Mar 25 18:22 myfile
-rwxrwxr-x 1 user user 51 Apr 9 14:42 script.sh
-rw-rw-r-- 1 user user 2 Mar 29 08:11 strings.txt
-rw-rw-r-- 1 root root 0 Mar 25 18:52 txt
-rw-rw-r-- 1 root root 79 Mar 29 13:52 user.txt
-rw-rw-r-- 1 root root 51 Mar 29 08:43 words
```



Права доступа



Права доступа





Права доступа

- ✓ **SUID** – запуск программы со своими разрешениями (s в пользовательских разрешениях запуска, 4 – для числового)
- ✓ **SGID** – запуск программы или переход в каталог с разрешениями группы владельца (s в разрешениях группы запуска, 2 – для числового)
- ✓ **Sticky bit** – права удаления только у владельца пользователя (t в разрешениях «других» на запуск, 1 – для числового)



Права доступа

u – права пользователя			g – права группы			o – права «других»		
r	w	x	r	w	x	r	w	x
4	2	1	4	2	1	4	2	1

```
[student@MSKtraining-35 ~]$ ls -la
total 244
drwx----- 9 student student 4096 Nov  1 14:33 .
drwxr-xr-x  3 root    root    4096 Oct 25 16:58 ..
-rw-r--r--  1 student student 85399 Oct 30 13:55 all-in-one-dbless-k4k8s-enterprise.yaml
-rw-r--r--  1 student student 85399 Oct 30 14:17 all-in-one-dbless-k4k8s-enterprise.yaml.1
-rw-----  1 student student 1268 Oct 30 19:32 .bash_history
-rw-----  1 student student  217 Jul 26  2021 .bash_logout
-rw-----  1 student student  259 Jul 26  2021 .bash_profile
-rw-----  1 student student  188 Jul 26  2021 .bashrc
drwx-----  2 student student 4096 Jul 26  2021 .cache
drwx-----  3 student student 4096 Oct 30 17:06 .config
-rw-----  1 student student  42 Oct 30 13:37 .lesshst
drwx-----  3 student student 4096 Oct 25 16:25 .local
-rw-----  1 student student  17 Jul 26  2021 .lpoptions
drwx-----  3 student student 4096 Oct 25 16:25 .mutt
drwxr-xr-x  2 student student 4096 Oct 30 17:03 new_dir
-rw-----  1 student student  121 Jul 26  2021 .rpmmacros
drwx-----  2 student student 4096 Oct 25 16:25 .ssh
-rw-r--r--  1 student student  0 Nov  1 14:22 test-file
-rw-r--r--  1 root    root    0 Nov  1 14:33 test-file-adm
-rw-r--r--  1 student student  780 Oct 30 14:17 test.yaml
-rw-r--r--  1 student student  180 Oct 30 14:17 .wget-hsts
-rwx-----  1 student student  319 Jul 26  2021 .xprofile
drwx-----  2 student student 4096 Jul 26  2021 .xsession.d
```



Владение. Команда chown и chgrp

ТОЛЬКО владелец может изменить владельца и права доступа (ну еще root)

<code>chown (change owner) <КТО?>:<КАКАЯ ГРУППА> <ЧТО></code>	#общий синтаксис
<code>chown petrov /opt/data/file1</code>	#пользователь petrov - новый владелец file1
<code>chown petrov:new_group /opt/data/file1</code>	#petrov - владелец file1, new_group - группа
<code>chown :new_group /opt/data/file1</code>	#изменение только группы
<code>chown -R :new_group /opt/data/</code>	#изменение группы для каталога и всего содержимого
<code>chgrp new_group /opt/data/file</code>	#изменение группы для file
<code>chgrp -R new_group /opt/data/</code>	#изменение группы для каталога и всего содержимого



Права доступа. Команда chmod

chmod (change mode) <ЧТО?> <ЧЕМУ?>

#общий синтаксис

chmod 765 /opt/data/file1

#user r+w+x group r+w other r+x

chmod u=rwx g=rw o=rw /opt/data/file1

#ЗАМЕНЯЕТ user r+w+x group r+w other r+x

chmod 777 /opt/data/file1

#user r+w+x group r+w+x other r+w+x

chmod +x /opt/data/file1

#ДОБАВЛЯЕТ user x group x other x

chmod a+x /opt/data/file1

#Добавляет **всем** x

chmod u+rwx,g+rw,o-x /opt/data/file1

#ДОБАВЛЯЕТ user r+w+x group r+w other -x

chmod u+s /opt/data/file1

#Добавляет user **SUID**

chmod g+s /opt/data/file1

#Добавляет group **SGID**

chmod g+t /opt/data/

#Добавляет group **sticky-бит** для каталога

chmod -R 755 /opt/data/

#Заменяет права каталогу и всему

содержимому

chmod --reference=/opt/data/file1 file2

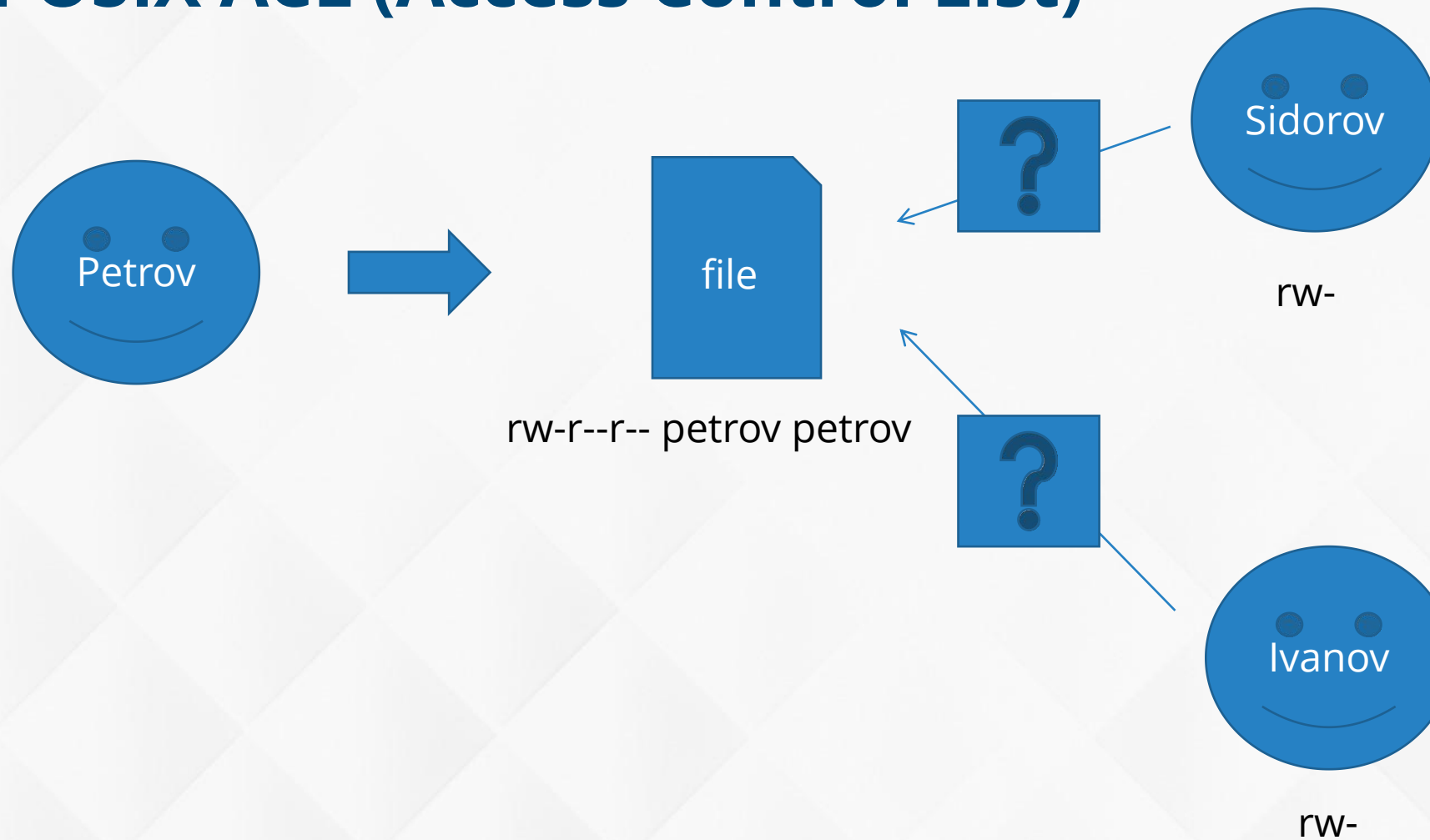
#КОПИРУЕТ права от file1 и устанавливает такие же file2

chmod 4765 /opt/data/file1

#user r+w+x group r+w other r+x



POSIX ACL (Access Control List)





Права доступа. POSIX ACL (Access Control List)

```
getfacl <имя_каталога_или_файла> #просмотр прав доступа

setfacl -m <кто>:<имя>:<права> <имя_файла> #добавление прав доступа

setfacl -m u:petrov:7 file1 #добавление пользователю petrov права rwx

setfacl -m g:group:rw- file1 #добавление группе group права rw

setfacl -m o:r-- file1 #добавление остальным права x

setfacl -x u:petrov file1 #удаление пользователя petrov из ACL прав

setfacl -x g:group file1 #удаление группы group из ACL прав

setfacl -b file1 #удаление всех ACL прав

setfacl -d -m u::rw,g::r,u:petrov:rwx /opt/data/ #ACL на каталог, которые
будут применяться для НОВЫХ файлов в данной директории
```

Лабораторная работа «Права доступа»



Модуль 12 . Архивация и компрессия





Архиватор tar

```
tar <опции> <file1> <file2> <dir1>
```

ОПЦИИ

```
-c  
-v  
-f  
-t  
-x
```

```
tar cfv archive.tar dir1 file1
```

```
tar tf archive.tar
```

```
tar xfv archive.tar
```

#общий синтаксис

#создание нового архива

#вывести список файлов в архиве

#упаковать в один архив

#просмотр архива

#распаковать архив

#file1 и dir упаковать в один архив archive.tar

#просмотр архива archive.tar

#распаковать архив archive.tar

Сжатие



```
gzip <опции> <file1>
```

ОПЦИИ

```
-k
```

```
-v
```

```
-d
```

```
-c
```

```
gzip file1
```

```
gzip -k file1
```

```
gzip -d file1.gz
```

```
tar cfvz archive.tar.gz dir1 file1
```

архив

```
tar cfvj archive.tar.gz dir1 file1
```

архив

#общий синтаксис

#сохранить исходный файл

#подробный вывод

#распаковать

#сохранить исходный файл и направить на STDOUT

#сжать и удалить file1

#сжать и сохранить file1

#распаковать file1.gz

#file1 и dir упаковать и сжать (gzip) в один archive.tar.gz

#file1 и dir упаковать и сжать (bzip2) в один archive.tar.bz2

Модуль 13 . Выполнение задач по расписанию



Cron



`minute(s) hour(s) day(s) month(s) weekday(s) command(s)`

`m h dom mon dow command`

минуты часы число месяц день_недели команда

минуты	0-59	#минута запуска
часы	0-23	#час запуска
число	1-31	#число месяца запуска
месяц	1-12	#месяц запуска
день_недели	0-6	#день недели запуска (0 - воскресенье)
команда		#команда или скрипт с указанием абсолютного пути

```
0 0 * * 0 /home/petrov.script.sh #запуск скрипта каждый понедельник в полночь
0 23 1 * * /home/petrov.script.sh #запуск скрипта 1 числа каждого месяца в 23:00
0 9,20 * * 1-5 /home/petrov.script.sh #в 9:00 и в 20:00 с понедельника по пятницу
*/20 * * * * /home/petrov.script.sh #запуск скрипта каждые 20 минут
```

Cron



```
crontab -e #редактирование расписания текущего пользователя
crontab -l #просмотр расписания текущего пользователя
crontab -r #удаление расписания текущего пользователя
crontab -u <имя_пользователя> -e #редактирование расписания другого пользователя

/var/spool/cron/crontabs/ #каталогс файлами расписаний всех пользователей
```

Лабораторная работа «Выполнение задач по расписанию»







Спасибо
за внимание!
Ваши вопросы...



Учебный центр «СПЕЦИАЛИСТ» – Ваш путь к успеху



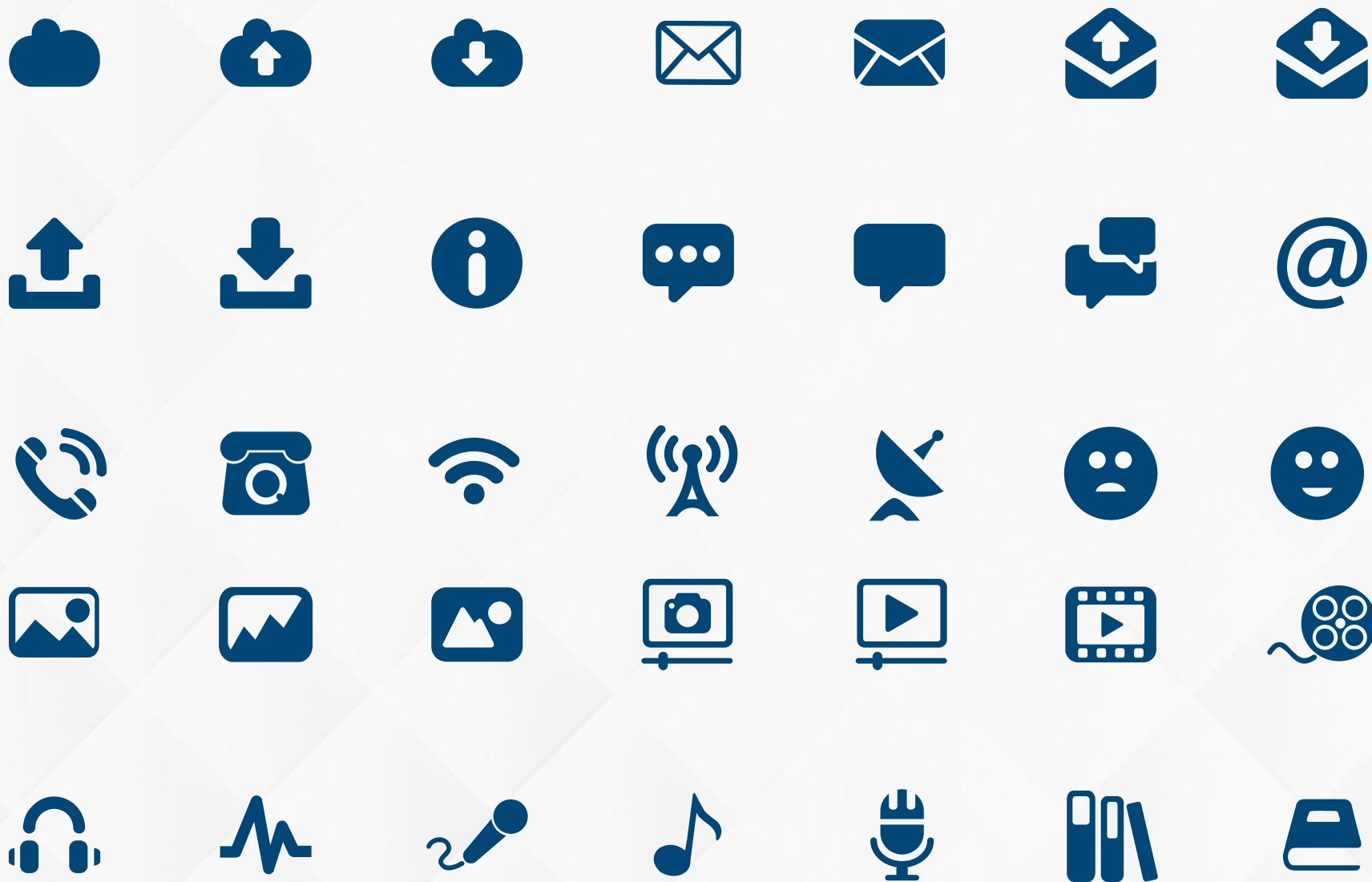
info@specialist.ru

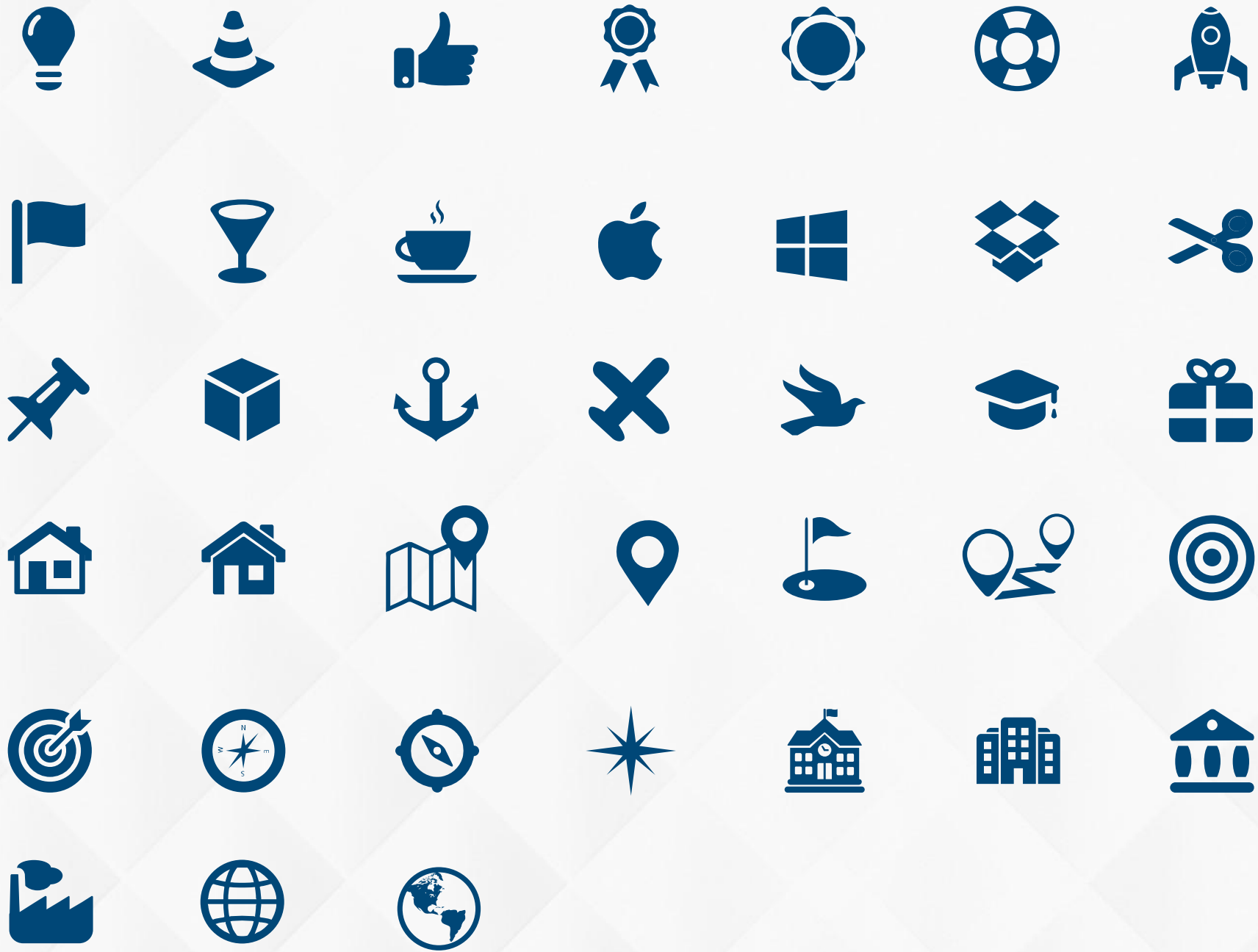


+7 (495) 232-32-16















ЭТОТ КОНТЕНТ КУПЛЕН НА САЙТЕ **SKLADCHIK.ORG**



Платформа

Это платформа, где каждый день тысячи людей собираются вместе, чтобы общими усилиями находить, приобретать и изучать курсы интересные именно им.



Сообщество

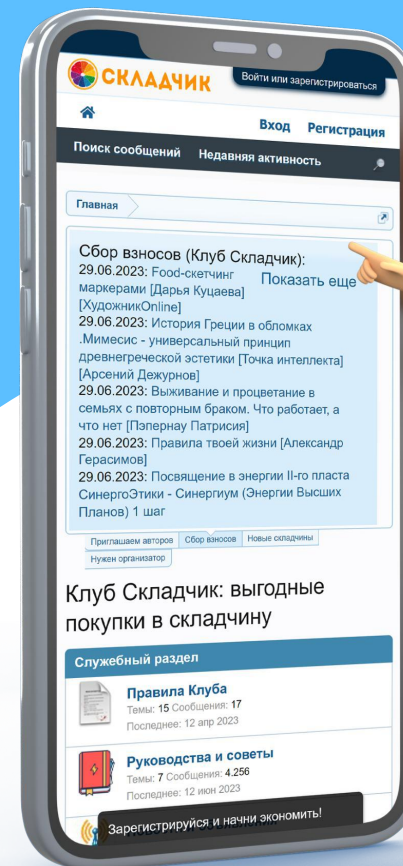
Это сообщество из 500 000 людей, открывших для себя выгодный способ быть в тренде самых актуальных и получить ценные знания по минимальной цене.



Библиотека

Это крупнейшая библиотека инфопродуктов в которой можно найти практически любой курс или тренинг продававшийся за последние 10 лет, а также уникальные авторские инфопродукты, которые не получится найти больше нигде.

ЧТО ТАКОЕ КЛУБ «СКЛАДЧИК»?



**ТЫ ЕЩЕ
НЕ В КЛУБЕ?**
ПРИСОЕДИНЯЙСЯ
И НАЧИНАЙ ЭКОНОМИТЬ!

