

# Сети экзамен по пунктам

## ▼ 1

### ▼ ISP

1. hostnamectl set-hostname isp
2. nano /etc/net/sysctl.conf
  - a. net.ipv4.ip\_forward = 1
3. cd /etc/net/ifaces
4. cp -r ens18/ ens19
5. vim ens19/options
  - a. BOOTPROTO=static
6. cp -r ens19/ ens20
7. cp -r ens19/ ens21
8. vim ens19/ipv4address
  - a. 100.100.100.1/28
9. vim ens19/ipv4route
  - a. 10.10.10.0/24 via 100.100.100.10
10. vim ens20/ipv4address
  - a. 150.150.150.1/28
11. vim ens20/ipv4route
  - a. 20.20.20.0/24 via 150.150.150.10
12. vim ens21/ipv4address
  - a. 35.35.35.1/28
13. systemctl restart network
14. reboot
15. apt-get update && apt-get install nftables chrony -y
16. vim /etc/nftables/nftables.nft

- a. в начало:  
flush ruleset;
- b. в конец:  
table ip nat {  
 chain postrouting {  
 type nat hook postrouting priority 0;  
 oifname ens18 masquerade;  
 }  
}
- c. Вот так:

```
flush ruleset

table inet filter {
  chain input {
    type filter hook input priority 0;
  }
  chain forward {
    type filter hook forward priority 0;
  }
  chain output {
    type filter hook output priority 0;
  }
}

table ip nat {
  chain postrouting {
    type nat hook postrouting priority 0;
    oifname enp0s3 masquerade;
  }
}
```

17. systemctl enable --now nftables
18. nft -f /etc/nftables/nftables.nft

▼ CLI

1. hostnamectl set-hostname cli
2. cd /etc/net/ifaces
3. cp -r ens18/ ens19
4. vim ens19/options
  - a. BOOTPROTO=static
5. vim ens19/ipv4address
  - a. 35.35.35.10/28
6. vim ens19/ipv4route
  - a. default via 35.35.35.1
7. systemctl restart network
8. reboot
9. apt-get update && apt-get install chrony yandex-browser -y

▼ RTR-L

1. hostnamectl set-hostname rtr-l
2. vim /etc/net/sysctl.conf
  - a. net.ipv4.ip\_forward = 1
3. cd /etc/net/ifaces
4. vim ens18/options
  - a. BOOTPROTO=static
5. cp -r ens18/ ens19
6. vim ens18/ipv4address
  - a. 100.100.100.10/28
7. vim ens18/ipv4route
  - a. default via 100.100.100.1
8. vim ens19/ipv4address
  - a. 10.10.10.1/24
9. systemctl restart network
10. reboot
11. apt-get update && apt-get install nftables chrony strongswan -y

▼ RTR-R

1. hostnamectl set-hostname rtr-r
2. vim /etc/net/sysctl.conf
  - a. net.ipv4.ip\_forward = 1
3. cd /etc/net/ifaces
4. vim ens18/options
  - a. BOOTPROTO=static
5. cp -r ens18/ ens19
6. vim ens18/ipv4address
  - a. 150.150.150.10/28
7. vim ens18/ipv4route
  - a. default via 150.150.150.1
8. vim ens19/ipv4address
  - a. 20.20.20.1/24
9. systemctl restart network
10. reboot
11. apt-get update && apt-get install chrony nftables strongswan -y

▼ WEB-L

1. hostnamectl set-hostname web-l
2. cd /etc/net/ifaaces/ens18/
3. vim options
  - a. BOOTPROTO=static
4. vim ipv4address
  - a. 10.10.10.110/24
5. vim ipv4route
  - a. default via 10.10.10.1
6. systemctl restart network
7. reboot
8. apt-get update && apt-get install chrony docker-io docker-compose nfs-clients -y

▼ WEB-R

1. hostnamectl set-hostname web-r
2. cd /etc/net/ifaaces/ens18/
3. vim options
  - a. BOOTPROTO=static
4. vim ipv4address
  - a. 20.20.20.100/24
5. vim ipv4route
  - a. default via 20.20.20.1
6. systemctl restart network
7. reboot
8. apt-get update && apt-get install chrony bind bind-utils nfs-clients -y

▼ SRV-L

1. hostnamectl set-hostname srv-l
2. cd /etc/net/ifaaces/ens18/
3. vim options
  - a. BOOTPROTO=static
4. vim ipv4address
  - a. 10.10.10.100/24
5. vim ipv4route
  - a. default via 10.10.10.1
6. systemctl restart network
7. reboot
8. apt-get update && apt-get install chrony bind bind-utils nfs-server -y

▼ 2

▼ RTR-L

1. vim /etc/nftables/nftables.nft

- a. в начало:  
flush ruleset;
- b. в конец:  
table ip nat {  
  chain postrouting {  
    type nat hook postrouting priority 0;  
    ip saddr 10.10.10.0/24 oifname ens18 masquerade;  
  }  
  chain prerouting {  
    type nat hook prerouting priority 0;  
    tcp dport 2024 dnat to 10.10.10.110:2024;  
  }  
}
- c. Вот так

```
flush ruleset;

table inet filter {
  chain input {
    type filter hook input priority 0;
  }
  chain forward {
    type filter hook forward priority 0;
  }
  chain output {
    type filter hook output priority 0;
  }
}

table ip nat {
  chain postrouting {
    type nat hook postrouting priority 0;
    ip saddr 10.10.10.0/24 oifname ens18 masquerade;
  }
  chain prerouting {
    type nat hook prerouting priority 0;
    tcp dport 2024 dnat to 10.10.10.110:2024;
  }
}
```

2. systemctl enable --now nftables

3. nft -f /etc/nftables/nftables.nft

▼ RTR-R

1. vim /etc/nftables/nftables.nft

- a. в начало:  
flush ruleset;
- b. в конец:  
table ip nat {  
  chain postrouting {  
    type nat hook postrouting priority 0;  
    ip saddr 20.20.20.0/24 oifname ens18 masquerade;  
  }  
  chain prerouting {  
    type nat hook prerouting priority 0;  
    tcp dport 2024 dnat to 20.20.20.100:2024;  
  }  
}
- c. Вот так

```
flush ruleset;

table inet filter {
  chain input {
    type filter hook input priority 0;
  }
  chain forward {
    type filter hook forward priority 0;
  }
  chain output {
    type filter hook output priority 0;
  }
}

table ip nat {
  chain postrouting {
    type nat hook postrouting priority 0;
    ip saddr 20.20.20.0/24 oifname ens18 masquerade;
  }
  chain prerouting {
    type nat hook prerouting priority 0;
    tcp dport 2024 dnat to 20.20.20.100:2024;
  }
}
```

```
}
```

c. Bot tak:

```
flush ruleset;

table inet filter {
    chain input {
        type filter hook input priority 0;
    }
    chain forward {
        type filter hook forward priority 0;
    }
    chain output {
        type filter hook output priority 0;
    }
}

table ip nat {
    chain postrouting {
        type nat hook postrouting priority 0;
        ip saddr 20.20.20.0/24 oifname ens18 masquerade;
    }
    chain prerouting {
        type nat hook prerouting priority 0;
        tcp dport 2024 dnat to 20.20.20.100:2024;
    }
}
```

2. `systemctl enable --now nftables`

3. `nft -f /etc/nftables/nftables.nft`

▼ 3

▼ RTR-L

1. `vim /etc/gre.up`
  - a. `#!/bin/bash`  
`ip tunnel add tun0 mode gre local 100.100.100.10 remote 150.150.150.10`  
`ip addr add 10.5.5.1/30 dev tun0`  
`ip link set up tun0`  
`ip route add 20.20.20.0/24 via 10.5.5.2`
2. `chmod +x /etc/gre.up`
3. `/etc/gre.up`
4. `vim /etc/crontab`
  - a. в конец добавляем:  
`@reboot root /etc/gre.up`
5. `vim /etc/strongswan/ipsec.conf`
  - a. ниже "config setup" пишем:  
`conn vpn`  
(следующие строки через tab)  
`auto=start`  
`type=tunnel`  
`authby=secret`  
`left=100.100.100.10`  
`right=150.150.150.10`  
`leftsubnet=0.0.0.0/0`  
`rightsubnet=0.0.0.0/0`  
`leftprotoport=gre`  
`rightprotoport=gre`  
`ike=aes128-sha256-modp3072`  
`esp=aes128-sha256`

b. вот так

```
config setup
    # strictcrpolicies=yes
    # uniqueids = no
# Add connections here.
# Sample VPN connections
conn vpn
    auto=start
    type=tunnel
    authby=secret
    left=100.100.100.10
    right=150.150.150.10
    leftsubnet=0.0.0.0/0
    rightsubnet=0.0.0.0/0
    leftprotoport=gre
    rightprotoport=gre
    ike=aes128-sha256-modp3072
    esp=aes128-sha256
```

6. `vim /etc/strongswan/ipsec.secrets`
  - a. `100.100.100.10 150.150.150.10 : PSK "P@ssw0rd"`
7. `systemctl enable --now ipsec.service`

▼ RTR-R

1. vim /etc/gre.up
  - a. `#!/bin/bash`  
`ip tunnel add tun0 mode gre local 150.150.150.10 remote 100.`  
`ip addr add 10.5.5.2/30 dev tun0`  
`ip link set up tun0`  
`ip route add 10.10.10.0/24 via 10.5.5.1`
2. `chmod +x /etc/gre.up`
3. `/etc/gre.up`
4. vim /etc/crontab
  - a. в конец добавляем:  
`@reboot root /etc/gre.up`
5. vim /etc/strongswan/ipsec.conf
  - a. ниже "config setup" пишем:  
`conn vpn`  
(следующие строки через tab)  
`auto=start`  
`type=tunnel`  
`authby=secret`  
`left=150.150.150.10`  
`right=100.100.100.10`  
`leftsubnet=0.0.0.0/0`  
`rightsubnet=0.0.0.0/0`  
`leftprotoport=gre`  
`rightprotoport=gre`  
`ike=aes128-sha256-modp3072`  
`esp=aes128-sha256`

b. вот так

```
config setup
    # strictcrlpolicy=yes
    # uniqueids = no
# Add connections here.
# Sample VPN connections
conn vpn
    auto=start
    type=tunnel
    authby=secret
    left=150.150.150.10
    right=100.100.100.10
    leftsubnet=0.0.0.0/0
    rightsubnet=0.0.0.0/0
    leftprotoport=gre
    rightprotoport=gre
    ike=aes128-sha256-modp3072
    esp=aes128-sha256
```

6. vim /etc/strongswan/ipsec.secrets
  - a. `100.100.100.10 150.150.150.10 : PSK "P@ssw0rd"`
7. `systemctl enable --now ipsec.service`

▼ 4

▼ WEB-L

1. `vim /etc/openssh/banner.txt`
  - a. Authorized access only
2. `vim /etc/openssh/sshd_config`
  - a. раскомментируем строчку Port 22  
пишем вместо 22 порт 2024
  - b. раскомментируем строчку MaxAuthTries 6  
пишем вместо 6 попыток 2
  - c. раскомментируем строчку Banner none  
вместо none пишем путь к banner.txt /etc/openssh/banner.txt
  - d. добавляем в конец  
AllowUsers sshuser
3. `adduser sshuser`
4. `passwd sshuser`
  - a. P@ssw0rd два раза
5. `systemctl restart sshd`

▼ WEB-R

1. `vim /etc/openssh/banner.txt`
  - a. Authorized access only
2. `vim /etc/openssh/sshd_config`
  - a. раскомментируем строчку Port 22  
пишем вместо 22 порт 2024
  - b. раскомментируем строчку MaxAuthTries 6  
пишем вместо 6 попыток 2
  - c. раскомментируем строчку Banner none  
вместо none пишем путь к banner.txt /etc/openssh/banner.txt
  - d. добавляем в конец  
AllowUsers sshuser
3. `adduser sshuser`
4. `passwd sshuser`
  - a. P@ssw0rd два раза
5. `systemctl restart sshd`



▼ 5

▼ SRV-L

1. systemctl enable --now bind
2. vim /etc/bind/options.conf

- a. что должно быть в options:
 

```
listen-on { any; };
forwarders { 94.232.137.104; };
dnssec-validation no;
recursion yes;
allow-query { any; };
allow-recursion { any; };
```
- b. вот так

```
options {
    version "unknown";
    directory "/etc/bind/zone";
    dump-file "/var/run/named_dump.db";
    statistics-file "/var/run/named.stats";
    recursing-file "/var/run/recursing";
    // disables the use of a PID file
    pid-file none;
    listen-on { any; };
    forwarders { 94.232.137.104; };
    dnssec-validation no;
    recursion yes;
    allow-query { any; };
    allow-recursion { any; };
}
```

3. vim /etc/bind/local.conf

- a. добавляем после слов Add other zones here:
 

```
zone "au.team" {
    type master;
    file "au.team";
    allow-transfer {20.20.20.100;};
};

zone "10.10.10.in-addr.arpa" {
    type master;
    file "left.reverse";
    allow-transfer {20.20.20.100;};
};

zone "20.20.20.in-addr.arpa" {
    type master;
    file "right.reverse";
    allow-transfer {20.20.20.100;};
};

zone "35.35.35.in-addr.arpa" {
    type master;
    file "cli.reverse";
    allow-transfer {20.20.20.100;};
};
```
- b. вот так

```
zone "au.team" {
    type master;
    file "au.team";
    allow-transfer {20.20.20.100;};
};

zone "10.10.10.in-addr.arpa" {
    type master;
    file "left.reverse";
    allow-transfer {20.20.20.100;};
};

zone "20.20.20.in-addr.arpa" {
    type master;
    file "right.reverse";
    allow-transfer {20.20.20.100;};
};

zone "35.35.35.in-addr.arpa" {
    type master;
    file "cli.reverse";
    allow-transfer {20.20.20.100;};
};
```

4. cd /etc/bind/zone/
5. cp localhost au.team

6. vim au.team

- заменяем localhost. на au.team. и root.localhost. на root.au.team.
- редачим зоны, должны получиться такие зоны, пишем через табуляцию (Tab), а не пробелы:

```
@      IN      NS      au.team.
@      IN      A      10.10.10.100
isp     IN      A      100.100.100.1
rtr-1   IN      A      10.10.10.1
rtr-r   IN      A      20.20.20.1
web-1   IN      A      10.10.10.110
web-r   IN      A      20.20.20.100
srv-1   IN      A      10.10.10.100
cli     IN      A      35.35.35.10
dns     IN      CNAME  srv-1
ntp     IN      CNAME  isp
mediawiki IN    CNAME  web-1
```

- должно получиться так

```
$TTL      1D
@         IN      SOA      au.team. root.au.team. (
                                2024021400      ; serial
                                12H               ; refresh
                                1H               ; retry
                                1W               ; expire
                                1H               ; ncache
                                )
@         IN      NS      au.team.
@         IN      A      10.10.10.100
isp       IN      A      100.100.100.1
rtr-1     IN      A      10.10.10.1
srv-1     IN      A      10.10.10.100
web-1     IN      A      10.10.10.110
rtr-r     IN      A      20.20.20.1
web-r     IN      A      20.20.20.100
cli       IN      A      35.35.35.10
dns       IN      CNAME  srv-1
ntp       IN      CNAME  isp
mediawiki IN      CNAME  web-1
```

7. cp localhost right.reverse

8. vim right.reverse

- заменяем localhost. на 20.20.20.in-addr.arpa. и root.localhost. на root.20.20.20.in-addr.arpa.
- редачим зоны, должны получиться такие зоны пишем через табуляцию (Tab), а не пробелы:

```
@      IN      NS      au.team.
@      IN      A      20.20.20.100
1      PTR     rtr-r.au.team.
100    PTR     web-r.au.team.
```

- должно получиться так

```
$TTL      1D
@         IN      SOA      20.20.20.in-addr.arpa. root.20.20.20.in-addr.arpa. (
                                2024021400      ; serial
                                12H               ; refresh
                                1H               ; retry
                                1W               ; expire
                                1H               ; ncache
                                )
@         IN      NS      au.team.
@         IN      A      20.20.20.100
1         PTR     rtr-r.au.team.
100       PTR     web-r.au.team.
```

9. cp right.reverse left.reverse

# 10. vim left.reverse

- заменяем 20.20.20.in-addr.arpa. на 10.10.10.in-addr.arpa. и root.20.20.20.in-addr.arpa. на root.10.10.10.in-addr.arpa.
- редачим зоны, должны получиться такие зоны пишем через табуляцию (Tab), а не пробелы:

```
@      IN      NS      au.team.
@      IN      A       10.10.10.100
1      PTR     rtr-1.au.team.
100    PTR     srv-1.au.team.
110    PTR     web-1.au.team.
```

- должно получиться так

```
$TTL      1D
@      IN      SOA     10.10.10.in-addr.arpa. root.10.10.10.in-addr.arpa. (
                                2024021400      ; serial
                                12H               ; refresh
                                1H               ; retry
                                1W               ; expire
                                1H               ; ncache
                                )
@      IN      NS      au.team.
@      IN      A       10.10.10.100
1      PTR     rtr-1.au.team.
100    PTR     srv-1.au.team.
110    PTR     web-1.au.team.
```

# 11. cp right.reverse cli.reverse

# 12. vim cli.reverse

- заменяем 10.10.10.in-addr.arpa. на 35.35.35.in-addr.arpa. и root.10.10.10.in-addr.arpa. на root.35.35.35.in-addr.arpa.
- редачим зоны, должны получиться такие зоны пишем через табуляцию (Tab), а не пробелы:

```
@      IN      NS      au.team.
@      IN      A       35.35.35.1
1      PTR     isp.au.team.
10     PTR     cli.au.team.
```

- должно получиться вот так

```
$TTL      1D
@      IN      SOA     35.35.35.in-addr.arpa. root.35.35.35.in-addr.arpa. (
                                2024021400      ; serial
                                12H               ; refresh
                                1H               ; retry
                                1W               ; expire
                                1H               ; ncache
                                )
@      IN      NS      au.team.
@      IN      A       35.35.35.1
1      PTR     isp.au.team.
10     PTR     cli.au.team.
```

# 13. chmod 777 au.team

# 14. chmod 777 right.reverse

# 15. chmod 777 left.reverse

# 16. chmod 777 cli.reverse

# 17. systemctl restart bind

# 18. vim /etc/resolv.conf

- должен быть указан только один nameserver 127.0.0.1

```
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/interfaces/<interface>/resolv.conf instead.
#nameserver 8.8.8.8
#nameserver 192.168.30.73
nameserver 127.0.0.1
```

▼ WEB-R

1. systemctl enable --now bind
2. vim /etc/bind/options.conf

a. что должно быть в options:

```
listen-on { any; };
forwarders { 10.10.10.100; };
dnssec-validation no;
recursion yes;
allow-query { any; };
allow-recursion { any; };
```

b. вот так

```
options {
    version "unknown";
    directory "/etc/bind/zone";
    dump-file "/var/run/named_dump.db";
    statistics-file "/var/run/named.stats";
    recursing-file "/var/run/recursing";
    // disables the use of a PID file
    pid-file none;
    listen-on { any; };
    forwarders { 10.10.10.100; };
    dnssec-validation no;
    recursion yes;
    allow-query { any; };
    allow-recursion { any; };
```

3. vim /etc/bind/local.conf

a. добавляем после слов Add other zones here:

```
zone "au.team" {
    type slave;
    file "slave/au.team";
    masters {10.10.10.100;};
};
zone "10.10.10.in-addr.arpa" {
    type slave;
    file "slave/left.reverse";
    masters {10.10.10.100;};
};
zone "20.20.20.in-addr.arpa" {
    type slave;
    file "slave/right.reverse";
    masters {10.10.10.100;};
};
zone "35.35.35.in-addr.arpa" {
    type slave;
    file "slave/cli.reverse";
    masters {10.10.10.100;};
};
```

b. вот так:

```
zone "au.team" {
    type slave;
    file "au.team";
    master {10.10.10.100;};
};
zone "10.10.10.in-addr.arpa" {
    type slave;
    file "left.reverse";
    master {10.10.10.100;};
};
zone "20.20.20.in-addr.arpa" {
    type slave;
    file "right.reverse";
    master {10.10.10.100;};
};
zone "35.35.35.in-addr.arpa" {
    type slave;
    file "cli.reverse";
    master {10.10.10.100;};
};
```

4. chown named:named /var/lib/bind/zone/slave/
5. chown named:named /etc/bind/zone/slave/
6. systemctl restart bind

7. vim /etc/resolv.conf

- a. должен быть указан только один nameserver 127.0.0.1

```
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/resolv.conf instead.
#nameserver 8.8.8.8
#nameserver 192.168.30.73
nameserver 127.0.0.1
```

▼ CLI

1. vim /etc/resolv.conf

- a. должен быть указан только один nameserver 10.10.10.100

```
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/
#nameserver 192.168.30.73
nameserver 10.10.10.100
```

▼ ISP

1. vim /etc/resolv.conf

- a. должен быть указан только один nameserver 10.10.10.100

```
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/
#nameserver 192.168.30.73
nameserver 10.10.10.100
```

▼ RTR-L

1. vim /etc/resolv.conf

- a. должен быть указан только один nameserver 10.10.10.100

```
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/
#search au.team
nameserver 10.10.10.100
#nameserver 172.16.3.20
#nameserver 172.16.3.21
```

▼ RTR-R

1. vim /etc/resolv.conf

- a. должен быть указан только один nameserver 20.20.20.100 (если WEB-R не работает, то 10.10.10.100)

```
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/
#search au.team
nameserver 20.20.20.100
#nameserver 172.16.3.20
#nameserver 172.16.3.21
```

▼ WEB-L

1. vim /etc/resolv.conf

- a. должен быть указан только один nameserver 10.10.10.100

```
# Generated by resolvconf
# Do not edit manually, use
# /etc/net/ifaces/<interface>/
#search au.team
nameserver 10.10.10.100
#nameserver 172.16.3.20
#nameserver 172.16.3.21
```

▼ 6

▼ ISP

1. vim /etc/chrony.conf
  - a. в конец пишем:  
server 127.0.0.1  
allow 100.100.100.0/28  
allow 150.150.150.0/28  
allow 35.35.35.0/28  
allow 10.10.10.0/24  
allow 20.20.20.0/24  
local stratum 5
2. systemctl restart chronyd

▼ CLI

1. vim /etc/chrony.conf
  - a. комментируем (пишем #) перед "pool pool.ntp.org iburst"
  - b. в конец пишем:  
server 35.35.35.1 iburst
2. systemctl restart chronyd

▼ RTR-L

1. vim /etc/chrony.conf
  - a. комментируем (пишем #) перед "pool pool.ntp.org iburst"
  - b. в конец пишем:  
server 100.100.100.1 iburst
2. systemctl restart chronyd

▼ RTR-R

1. vim /etc/chrony.conf
  - a. комментируем (пишем #) перед "pool pool.ntp.org iburst"
  - b. в конец пишем:  
server 150.150.150.1 iburst
2. systemctl restart chronyd

▼ WEB-R

1. vim /etc/chrony.conf
  - a. комментируем (пишем #) перед "pool pool.ntp.org iburst"
  - b. в конец пишем:  
server 150.150.150.1 iburst
2. systemctl restart chronyd

▼ WEB-L

1. vim /etc/chrony.conf
  - a. комментируем (пишем #) перед "pool pool.ntp.org iburst"
  - b. в конец пишем:  
server 100.100.100.1 iburst
2. systemctl restart chronyd

▼ SRV-L

1. vim /etc/chrony.conf
  - a. комментируем (пишем #) перед "pool pool.ntp.org iburst"
  - b. в конец пишем:  
server 100.100.100.1 iburst
2. systemctl restart chronyd

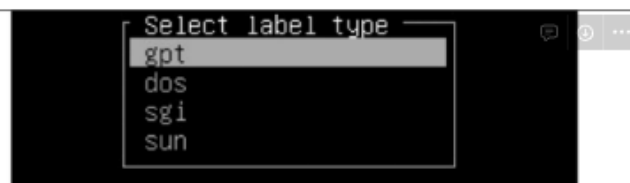
▼ 7

▼ SRV-L

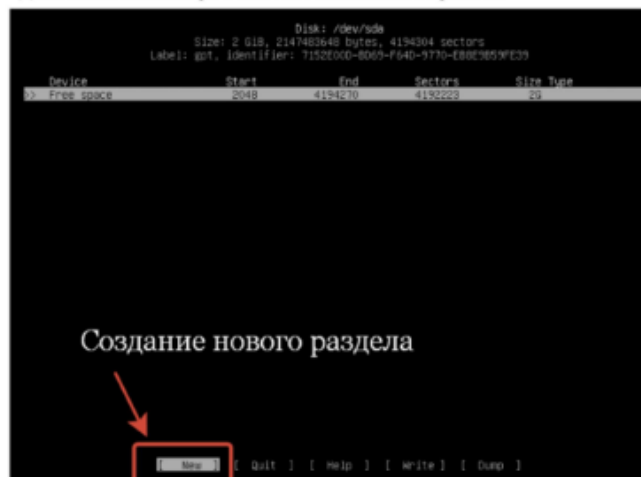
1. lsblk проверяем NAME 4 дисков размером 1G:  
в моем случае 4 диска размером 1 гб это sdb sdc sdd sde  
ДОЛЖНО БЫТЬ 4 ДИСКА: sdb sdc sdd sde

```
(root@srv-l ~)# lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
sda          8:0    0   40G  0 disk
├─sda1       8:1    0    1G  0 part [SWAP]
└─sda2       8:2    0   38,1G 0 part /
sdb          8:16   0    1G  0 disk
sdc          8:32   0    1G  0 disk
sdd          8:48   0    1G  0 disk
sde          8:64   0    1G  0 disk
sr0         11:0    1 1024M  0 rom
```

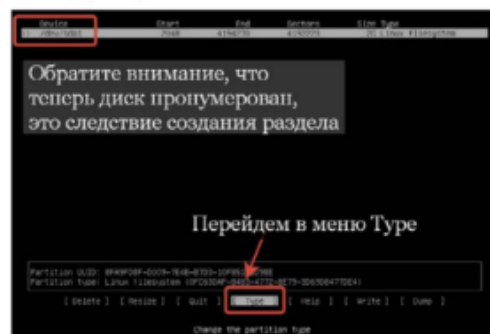
2. cfdisk /dev/sdb
3. List



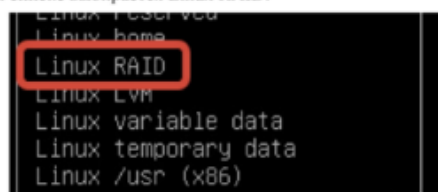
Далее необходимо выбрать New для создания нового раздела.



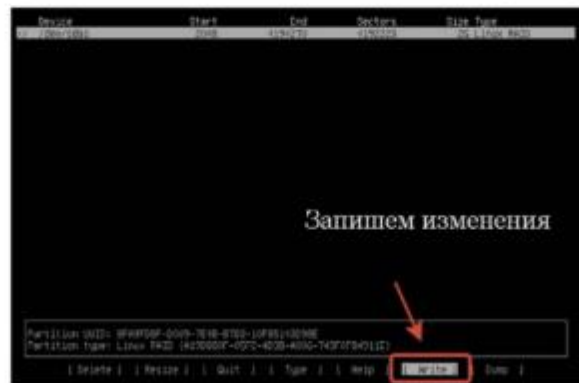
4. Enter
5. List



В указанном списке выбирается Linux RAID.



## 6. List



Вручную пишется слово yes, чтобы принять изменения и отформатировать диск.

```
Are you sure you want to write the partition table to disk? yes
```

Type "yes" or "no", or press ESC to leave this dialog.

## 7. List

После внесенных правок необходимо покинуть утилиту cfdisk, выбрав параметр Quit.

```
Partition UUID: 8FA9F08F-0009-7E4B-B703-10F85143D98E
Partition type: Linux RAID (A190680F-05FC-403B-A006-743F0F84911E)

[ Delete ] [ Resize ] [ Quit ] [ Type ] [ Help ] [ Write ] [ Dump ]
```

8. С пункта 2 повторить действия со всеми остальными дисками (sdb, sdc, sdd, sde)
9. `mdadm --create /dev/md0 --level=5 --raid-devices=4 /dev/sdb1 /dev/sdc1 /dev/sdd1 /dev/sde1`
10. `mdadm --detail --scan --verbose | tee -a /etc/mdadm.conf`
11. `mkfs.ext4 /dev/md0`
12. `mkdir /raid5`
13. `vim /etc/fstab`
  - a. добавить в конец, пишем через табуляцию, а не пробел:  
`/dev/md0 /raid5 ext4 defaults 0 0`
  - b. так

```
proc                /proc                proc                nosuid,noexec,gid=proc
devpts              /dev/pts             devpts             nosuid,noexec,gid=tty,mode=620
tmpfs               /tmp                 tmpfs              nosuid
UUID=9e08571f-9b6f-4f61-8425-782ad628950f /                    ext4               relatime
UUID=c04e0177-5eb9-4410-8685-3396f4271cb4 swap                 swap               defaults
/dev/md0             /raid5               ext4               defaults
0                    0
```

14. `reboot`
15. `mkdir /raid5/nfs`
16. `chmod 777 /raid5/nfs`
17. `vim /etc/exports`
  - a. в конец добавляем:  
`/raid5/nfs 10.10.10.110(rw,sync) 20.20.20.100(rw,sync)`



▼ WEB-L

1. `mkdir /mnt/nfs`
2. `vim /etc/fstab`
  - а. добавляем в конец, пишем через табуляцию, а не пробел:  
`10.10.10.100:/raid5/nfs /mnt/nfs nfs rw,sync 0 0`
3. `mount -av`

▼ WEB-R

1. `mkdir /mnt/nfs`
2. `vim /etc/fstab`
  - а. добавляем в конец, пишем через табуляцию, а не пробел:  
`10.10.10.100:/raid5/nfs /mnt/nfs nfs rw,sync 0 0`
3. `mount -av`

▼ 8

▼ WEB-L

1. `systemctl disable --now ahttpd`  
`systemctl disable --now alerterd`

2. `vim ~/wiki.yml`

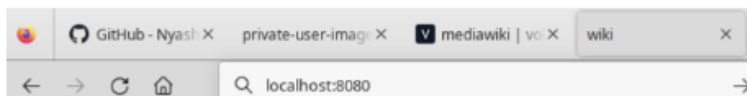
а. пишем это:

```
version: '3'
services:
  Mediawiki:
    container_name: wiki
    image: mediawiki
    restart: always
    ports:
      - 8080:80
    links:
      - database
    volumes:
      - images:/var/www/html/images
      # - ./LocalSettings.php:/var/www/html/LocalSettings.php

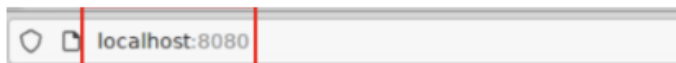
  database:
    container_name: db
    image: mysql
    restart: always
    environment:
      MYSQL_DATABASE: mediawiki
      MYSQL_USER: wiki
      MYSQL_PASSWORD: DEP@ssw0rd
      MYSQL_RANDOM_ROOT_PASSWORD: 'toor'
    volumes:
      - dbvolume:/var/lib/mysql

volumes:
  images:
  dbvolume:
    external: true
```

3. `systemctl enable --now docker`
4. `docker volume create dbvolume`
5. `cd ~`
6. `docker-compose -f wiki.yml up -d`
7. заходим в mozilla, пишем в строке url:  
`localhost:8080`



8. жмем set up the wiki



**MediaWiki 1.41.0**

LocalSettings.php not found.

Please [set up the wiki](#) first.

Установка MediaWiki 1.41.0

10. далее

Авторские права и условия

MediaWiki — свободное программное обеспечение, которое вы можете распространять и/или изменять опубликованной фондом свободного программного обеспечения; второй версии, либо любой более поздней версии MediaWiki распространяется в надежде, что она будет полезной, но **без каких-либо гарантий**, даже без пригодности для определённой цели. См. лицензию GNU General Public License для более подробной информации. Мы должны были получить копию GNU General Public License вместе с этой программой, если нет, то на Floor, Boston, MA 02110-1301, USA или прочитайте ее онлайн.

11. Пароль: DEP@ssw0rd

**тип базы данных:**

☒ MariaDB, MySQL или совместимая

☐ SQLite

**Настройки MariaDB/MySQL**

**Хост базы данных:**

☐ [справка](#)

☐ Подключиться через SSL

**Идентификация этой вики**

**Имя базы данных (без дефисов):**

☐ [справка](#)

**Префикс таблиц базы данных (без дефисов):**

☐ [справка](#)

**Учётная запись для установки**

**Имя пользователя базы данных:**

☐ [справка](#)

**Пароль базы данных:**

☐ [справка](#)

12. Далее

### Настройки базы данных

Учётная запись для доступа к базе данных из веб-сервера

☒ Использовать ту же учётную запись, что и для установки

[← Назад](#) [Далее →](#)

13. List

### Название

Название вики:

[справка](#)

Пространство имён проекта:

[справка](#)

☒ То же, что имя вики: Wiki

☐ Проект

☐ Другое (укажите)

Учётная запись администратора

Ваше имя участника:

[справка](#)

14. Пароль: DEP@ssw0rd почту не указываем

### Пароль:

Пароль ещё раз:

Адрес электронной почты:

[справка](#)

[справка](#)

☐ Подписаться на рассылку новостей о появлении новых версий MediaWiki.

[справка](#)

☒ Поделитьсь сведениями об этой установке с разработчиком MediaWiki.

[Политика конфиденциальности.](#)

Вы почти у цели! Остальные настройки можно пропустить и приступить к установке вики.

☐ Произвести тонкую настройку

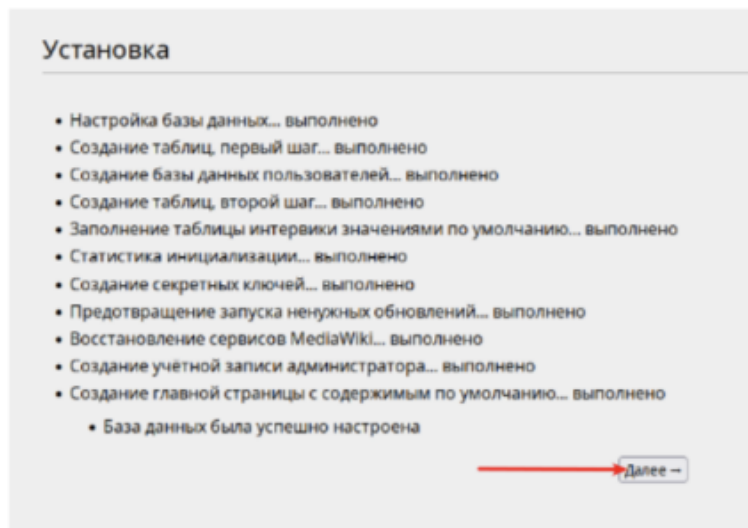
☒ Хватит, установить вики

[← Назад](#) [Далее →](#)

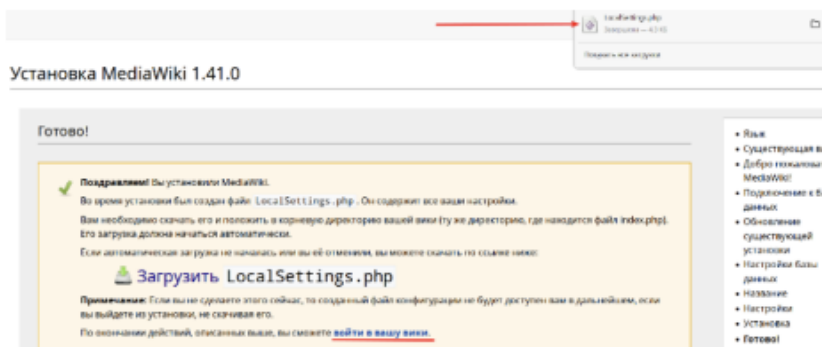
15. Далее

### Установка

15. Далее



16. Жмем до конца далее и скачается файл, надо найти куда этот файл скачался, скорее всего вот сюда /home/user/Загрузки/



17. Копируем скачанный файл:

```
cp /home/user/Загрузки/LocalSettings.php ~/LocalSettings.php
```

18. vim ~/wiki.yml

- a. прокомментируем  
- ./LocalSettings.php/var/www/html/LocalSettings.php

19. vim ~/LocalSettings.php

- a. \$wgServer = "http://mediawiki.au.team:8080"

20. docker-compose -f wiki.yml stop

21. docker-compose -f wiki.yml up -d

#### ▼ WEB-R

1. systemctl disable --now ahttpd  
systemctl disable --now altermtd

#### ▼ 9

##### ▼ CLI

1. apt-get install yandex-browser -y
2. запустить HE от рута с помощью команды:  
yandex-browser-stable  
запустить от рута с помощью команды:  
yandex-browser-stable --no-sandbox

##### ▼ Трешшутинг

Если DNS сервер не работает, systemctl status bind выдает ошибки, надо systemctl restart bind на DNS сервере