МОДУЛЬ В

⏱️ ТАЙМИНГ НА ЧЕМПИОНАТЕ (4 ЧАСА)

РЕКОМЕНДОВАННЫЙ ПЛАН:

```
00:00-00:10 (10 мин) — Изучить задание, топологию, записать I
P
00:10-00:25 (15 мин) — SSH + Fail2ban на всех серверах
00:25-00:50 (25 мин) — FreeIPA: создать 71 пользователя
00:50-01:10 (20 мин) — Парольная политика + Petya + backup
01:10-01:40 (30 мин) — OpenVPN между роутерами
01:40-02:00 (20 мин) — OSPF
02:00-02:30 (30 мин) — HTTPS веб-серверы
02:30-02:50 (20 мин) — NFS + PC1 в домен
02:50-03:10 (20 мин) — NFTables
03:10-03:30 (20 мин) — Wireguard + автопереключение
03:30-03:50 (20 мин) — Docker + поиск уязвимостей
03:50-04:00 (10 мин) — Проверка, черновик
```

1️⃣ ЗАЧЕМ ТОПОЛОГИЯ, ЕСЛИ ТЫ ВСЁ ЗНАЕШЬ?

РЕАЛЬНОСТЬ НА ЧЕМПИОНАТЕ:

ТОПОЛОГИЯ БУДЕТ ДРУГАЯ!

Что я имею в виду:

- IP-адреса могут быть **другими** (не 192.168.1.10, а 172.16.5.10)
- Названия серверов могут быть **другими** (не MAIN-DC1, а SRV-IPA-01)
- Количество серверов может **отличаться**
- Связи между ними могут быть **другими**

Поэтому тебе дадут схему, чтобы ты понял:

```
Какой сервер где находится?
Какие IP-адреса использовать?
Через какой роутер идёт трафик?
```

ПРИМЕР:

Ты выучил:

```
MAIN-DC1: 192.168.1.10
PC1: 192.168.3.20
```

На чемпионате может быть:

```
DC-PRIMARY: 10.50.1.5
CLIENT-01: 10.50.3.100
```

Топология покажет тебе настоящие адреса!

📄 ОТЧЁТНОСТЬ

ЧТО НУЖНО ОСТАВИТЬ:

1. Черновик (бумажный лист)

Что писать:

```
=== МОДУЛЬ B: Защита инфраструктуры ===

УЧЁТНЫЕ ДАННЫЕ:
- SSH пользователь: sshuser / P@ssw0rd
- FreeIPA admin: admin / <пароль из задания>
- Backup пароль: BackupPa$$
```

```
IP-АДРЕСА (если изменил):
- MAIN-DC1: 192.168.1.10
- COD-WEB: 10.10.20.100

ВАЖНЫЕ ФАЙЛЫ:
- Backup: /mnt/backup-user.backup
- Инструкция восстановления: ~/BACKUP_RESTORE.txt
- Сертификаты CA: /etc/ca/

VPN:
- OpenVPN порт: 1194
- Wireguard порт: 51820

УЯЗВИМОСТИ (найденные):
ПЛК сервер:
1. Открытый порт 8080 без пароля
2. Docker запущен от root
3. Nginx версия устаревшая

CI/CD скрипты:
1. Пароли в открытом виде в скрипте deploy.sh
2. Нет проверки ошибок (нет set -e)
3. Использование eval с пользовательским вводом
4. Sudo без ограничений
```

---

2. Инструкция по восстановлению backup

Файл: `~/BACKUP_RESTORE.txt` на MAIN-DC1

```
=== ИНСТРУКЦИЯ ПО ВОССТАНОВЛЕНИЮ BACKUP FreeIPA ===

Пароль архива: BackupPa$$

Шаг 1: Расшифровать
openssl enc -aes-256-cbc -d -pbkdf2 -in /mnt/backup-user.back
```

```
up -out backup.tar.gz -pass pass:"BackupPa$$"

Шаг 2: Распаковать
tar -xzf backup.tar.gz

Шаг 3: Восстановить
ldapadd -x -D "cn=Directory Manager" -W -f ldap_*.ldif

Выполнять на MAIN-DC1 от root
```

---

3. Комментарии в конфигах

Если делаешь что-то нестандартное — оставь комментарий:

```
# В /etc/openvpn/server.conf
# Порт изменён на 1195 из-за конфликта с другим сервисом
port 1195
```

🗂️ ПОЛНАЯ ШПАРГАЛКА: КОМАНДЫ

1️⃣ SSH + FAIL2BAN + AUDITD

Создать пользователя SSH:

```
useradd -m -s /bin/bash sshuser
passwd sshuser
```

Настроить SSH:

```
echo "PermitRootLogin no" >> /etc/ssh/sshd_config
echo "AllowUsers sshuser" >> /etc/ssh/sshd_config
systemctl restart sshd
```

Fail2ban:

```
apt install -y fail2ban

cat > /etc/fail2ban/jail.local << 'EOF'
[DEFAULT]
bantime = 1m
maxretry = 2

[sshd]
enabled = true
EOF

systemctl enable fail2ban
systemctl restart fail2ban
```

Auditd:

```
apt install -y auditd

cat > /etc/audit/rules.d/ssh.rules << 'EOF'
-w /usr/sbin/sshd -p x -k ssh_exec
-w /etc/ssh/sshd_config -p wa -k ssh_config
EOF

systemctl restart auditd
```

2️⃣ VPN (КРАТКО)

OpenVPN сервер (COD-RTR):

```
apt install -y openvpn easy-rsa
make-cadir ~/openvpn-ca
cd ~/openvpn-ca
./easyrsa init-pki
```

```
./easyrsa build-ca nopass
./easyrsa gen-req server nopass
./easyrsa sign-req server server
./easyrsa gen-dh
openvpn --genkey secret ta.key
```

Конфиг `/etc/openvpn/server.conf`**:**

```
port 1194
proto udp
dev tun
ca /root/openvpn-ca/pki/ca.crt
cert /root/openvpn-ca/pki/issued/server.crt
key /root/openvpn-ca/pki/private/server.key
dh /root/openvpn-ca/pki/dh.pem
tls-auth /root/openvpn-ca/ta.key 0
server 10.8.0.0 255.255.255.0
push "route 10.10.10.0 255.255.255.0"
push "route 10.10.20.0 255.255.255.0"
cipher AES-256-GCM
persist-key
persist-tun
```

Wireguard (резервный):

```
apt install -y wireguard
cd /etc/wireguard
wg genkey | tee private.key | wg pubkey > public.key
```

Конфиг `/etc/wireguard/wg0.conf`**:**

```
[Interface]
Address = 10.9.0.1/24
ListenPort = 51820
PrivateKey = <содержимое private.key>
```

```
[Peer]
PublicKey = <публичный ключ клиента>
AllowedIPs = 10.9.0.2/32
```

3️⃣ FreeIPA

Войти:

```
kinit admin
```

Создать группы:

```
ipa group-add Administrators
ipa group-add Worker
ipa group-add TopManager
```

Создать 71 пользователя (скрипт):

```
for i in {1..40}; do
  echo -e "P@ssw0rdAdmin\nP@ssw0rdAdmin" | ipa user-add user
$i --first=User --last=$i --password 2>/dev/null
  ipa group-add-member Administrators --users=user$i 2>/dev/n
ull
done

for i in {41..80}; do
  echo -e "P@ssw0rdWorker\nP@ssw0rdWorker" | ipa user-add use
r$i --first=User --last=$i --password 2>/dev/null
  ipa group-add-member Worker --users=user$i 2>/dev/null
done

for i in {100..130}; do
  echo -e "P@ssw0rdTop\nP@ssw0rdTop" | ipa user-add superuser
$i --first=SuperUser --last=$i --password 2>/dev/null
```

```
  ipa group-add-member TopManager --users=superuser$i 2>/dev/
null
done
```

Парольная политика:

```
ipa pwpolicy-mod --minlength=10 --minclasses=3 --history=4 --
maxlife=31 --lockouttime=900 --maxfail=3
```

Пользователь Petya:

```
echo -e "Pa\$\$w0rd2026-2027\nPa\$\$w0rd2026-2027" | ipa user
-add petya --first=Petya --last=Petrov --password
ipa user-mod petya --principal-expiration=20260515000000Z
```

Backup:

```
ldapsearch -x -D "cn=Directory Manager" -w "<пароль>" -b "dc=
reaskills,dc=cyber" -LLL > /tmp/backup.ldif
tar -czf /tmp/backup.tar.gz /tmp/backup.ldif
openssl enc -aes-256-cbc -salt -pbkdf2 -in /tmp/backup.tar.gz
-out /mnt/backup-user.backup -pass pass:"BackupPa\$\$"
```

4️⃣ OSPF

```
apt install -y frr
sed -i 's/ospfd=no/ospfd=yes/' /etc/frr/daemons

vtysh
configure terminal
router ospf
  ospf router-id 1.1.1.1
  network 10.8.0.0/24 area 0
  network 10.10.10.0/24 area 0
```

```
exit
interface tun0
  ip ospf authentication message-digest
  ip ospf message-digest-key 1 md5 secret_key
exit
exit
write memory
exit

systemctl restart frr
```

5️⃣ NFTABLES

NAT:

```
apt install -y nftables
nft add table ip nat
nft add chain ip nat postrouting { type nat hook postrouting
priority 100 \; }
nft add rule ip nat postrouting oifname "eth0" masquerade
```

Rate limit:

```
nft add table inet filter
nft add chain inet filter input { type filter hook input prio
rity 0 \; policy drop \; }
nft add rule inet filter input ct state established,related a
ccept
nft add rule inet filter input iif lo accept
nft add rule inet filter input tcp dport 22 accept
nft add rule inet filter input limit rate 100/second accept
```

Сохранить:

```
nft list ruleset > /etc/nftables.conf
systemctl enable nftables
```

6️⃣ NFS

Сервер (MAIN-STORAGE):

```
apt install -y nfs-kernel-server
mkdir -p /data
chmod 777 /data
echo "/data 192.168.3.0/24(rw,sync,no_root_squash)" >> /etc/e
xports
exportfs -ra
systemctl restart nfs-kernel-server
```

Клиент (PC1):

```
apt install -y nfs-common
mkdir -p /mnt/home
mount 192.168.2.10:/data /mnt/home
echo "192.168.2.10:/data /mnt/home nfs defaults 0 0" >> /etc/
fstab
```

7️⃣ HTTPS ВЕБ-СЕРВЕРЫ

Получить сертификат (на MAIN-DC1):

```
kinit admin
ipa-getcert request -K HTTP/www.reaskills.cyber -k /tmp/www.k
ey -f /tmp/www.crt -D www.reaskills.cyber
```

Nginx с HTTPS:

```
apt install -y nginx

cat > /etc/nginx/sites-available/default << 'EOF'
server {
    listen 443 ssl;
    server_name www.reaskills.cyber;

    ssl_certificate /etc/nginx/www.crt;
    ssl_certificate_key /etc/nginx/www.key;

    root /data_saver;
    autoindex on;
}
EOF

systemctl restart nginx
```