DHCP-

DHCP- . DHCP- , :

- default-router – IP- , ;
- domain-name – , (DNS);
- dns-server – , . .

1	IPv4/IPv6 DHCP-.	`esr(config)# ip dhcp-server [vrf <VRF>]`	<VRF> – VRF, DHCP-. 31 .
		`esr(config)# ipv6 dhcp-server [vrf <VRF>]`	
2	DSCP IP- DHCP- ( ).	`esr(config)# ip dhcp-server dscp <DSCP>`	<DSCP> – DSCP, [0..63]. : 61.
3	IPv4/IPv6- DHCP- .	`esr(config)# ip dhcp-server pool <NAME> [vrf <VRF>]`	<NAME> – IPv4/IPv6- DHCP-, 31 . <VRF> – VRF, IP- DHCP-. 31
		`esr(config)# ipv6 dhcp-server pool <NAME> [vrf <VRF>]`	
4	IPv4/IPv6- , IPv4 /IPv6-.	`esr(config-dhcp-server)# network <ADDR/LEN>`	<ADDR/LEN> – IP- , AAA.BBB.CCC.DDD/EE, AAA – DDD [0..255] EE [1..32].
		`esr(config-ipv6-dhcp-server)# network <IPV6-ADDR /LEN>`	<IPV6-ADDR/LEN> – IP- , X:X:X:X::X/EE, X [0..FFFF] EE [1..128].
5	IPv4/IPv6- , DHCP-.	`esr(config-dhcp-server)# address-range <FROM-ADDR>-<TO-ADDR>`	<FROM-ADDR> – IP- ; <TO-ADDR> – IP- , AAA.BBB.CCC.DDD, [0..255]. 32 IP-, .
		`esr(config-ipv6-dhcp-server)# address-range <FROM-ADDR>-<TO-ADDR>`	<FROM-ADDR> – IPv6- ; <TO-ADDR> – IP- ; X:X:X:X::X, [0..FFFF].

6	IPv4/IPv6- DHCP- ( ).	`esr(config-dhcp-server)# address <ADDR> {mac-address <MAC> | client-identifier <CI>}`	<ADDR> – IP- , AAA.BBB.CCC.DDD, [0..255]; <MAC> – - , IP-, XX:XX:XX:XX:XX:XX, [00..FF]. <CI> – DHCPOption61. : • HH:HH:HH:HH:HH:HH:HH: – mac- ; • STRING – 1 64 .
		`esr(config-ipv6-dhcp-server)# address <ADDR> mac-address <MAC>`	<IPV6-ADDR> – IPv6- , X:X:X:X::X, [0..FFFF]; <MAC> – - , IPv6-, XX:XX:XX:XX:XX:XX, [00..FF]
7	IPv4- , DHCP- , DHCP- 3.	`esr(config-dhcp-server)# default-router <ADDR>`	<ADDR> – IP- , AAA.BBB.CCC.DDD, [0..255]. 8 IP-, .
8	DNS- . DHCP- 15 ( ).	`esr(config-dhcp-server)# domain-name <NAME>`	<NAME> – DNS- , 255 .
		`esr(config-ipv6-dhcp-server)# domain-name <NAME>`	
9	IPv4/IPv6- DNS-. DHCP- 6 ( ).	`esr(config-dhcp-server)# dns-server <ADDR>`	<ADDR> – IP- DNS-, AAA.BBB.CCC.DDD, [0..255]. 8 IP-, .
		`esr(config-ipv6-dhcp-server)# dns-server <IPV6-ADDR>`	<IPV6-ADDR> – IPv6- DNS-, X:X:X:X::X, [0..FFFF]. 8 IPv6-, .
10	IP- ( ). DHCP- , , , .	`esr(config-dhcp-server)# max-lease-time <TIME>`	<TIME> – IP-, DD:HH:MM, : • DD – , [0..364]; • HH – , [0..23]; • MM – , [0..59] : 1
		`esr(config-ipv6-dhcp-server)# max-lease-time <TIME>`	
11	, IP- ( ). .	`esr(config-dhcp-server)# default-lease-time <TIME>`	<TIME> – IP-, DD:HH:MM, : • DD – , [0..364]; • HH – , [0..23]; • MM – , [0..59] : 12 .
		`esr(config-ipv6-dhcp-server)# default-lease-time <TIME>`	
12	(DHCP 60) ( ).	`esr(config)# ip dhcp-server vendor-class-id <NAME>`	<NAME> – , 31 .
		`esr(config)# ipv6 dhcp-server vendor-class-id <NAME>`	
13	(DHCP 43).	`esr(config-dhcp-vendor-id)# vendor-specific-options <HEX>`	<HEX> – , 128 .
		`esr(config-ipv6-dhcp-vendor-id)# vendor-specific-options <HEX>`	
14	IP- NetBIOS- (DHCP 44) ( ).	`esr(config-dhcp-server)# netbios-name-server <ADDR>`	<ADDR> – IP- NetBIOS- AAA.BBB.CCC.DDD, [0..255]. 4 IP-.
15	IP- tftp- (DHCPOption 150) ( ).	`esr(config-dhcp-server)# tftp-server <ADDR>`	<ADDR> – IP- DNS-, AAA.BBB.CCC.DDD, [0..255].

:

DHCP- , «trusted». IP- 192.168.1.0/24 . 1 . , DNS- DHCP-.

:

«trusted» :

```
esr# configure
esr(config)# security zone trusted
esr(config-zone)# exit
```

«Simple» IP- . , , :

```
esr# configure
esr(config)# ip dhcp-server pool Simple
esr(config-dhcp-server)# network 192.168.1.0/24
esr(config-dhcp-server)# address-range 192.168.1.100-192.168.1.125
esr(config-dhcp-server)# default-lease-time 1:00:00
```

:

- : 192.168.1.1;
- : eltex.loc;
- DNS-: DNS1: 172.16.0.1, DNS2: 8.8.8.8.

```
esr(config-dhcp-server)# domain-name "eltex.loc"
esr(config-dhcp-server)# default-router 192.168.1.1
esr(config-dhcp-server)# dns-server 172.16.0.1,8.8.8.8
esr(config-dhcp-server)# exit
```

DHCP- IP- , IP-, , .

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# security-zone trusted
esr(config-if-gi)# ip address 192.168.1.1/24
esr(config-if-gi)# exit
```

DHCP , 68 67, DHCP, UDP:

```
esr(config)# object-group service dhcp_server
esr(config-object-group-service)# port-range 67
esr(config-object-group-service)# exit
esr(config)# object-group service dhcp_client
esr(config-object-group-service)# port-range 68
esr(config-object-group-service)# exit
esr(config)# security zone-pair trusted self
esr(config-zone-pair)# rule 30
esr(config-zone-rule)# match protocol udp
esr(config-zone-rule)# match source-port dhcp_client
esr(config-zone-rule)# match destination-port dhcp_server
esr(config-zone-rule)# action permit
esr(config-zone-rule)# enable
esr(config-zone-rule)# exit
esr(config-zone-pair)# exit
```

:

```
esr(config)# ip dhcp-server
esr(config)# exit
```

:

```
esr# show ip dhcp binding
```

:

```
esr# show ip dhcp server pool
esr# show ip dhcp server pool Simple
```

IPv6 IPv4.

Destination NAT

Destination NAT (DNAT) IP- , .

DNAT , «» , «» , . , .

1	.	`esr(config)# nat destination`	
2	C IP- / TCP/UDP- ( ).	`esr(config-dnat)# pool <NAME>`	<NAME> – NAT-, 31 .
3	IP-, IP- .	`esr(config-dnat-pool)# ip address <ADDR>`	<ADDR> – IP-, AAA.BBB.CCC.DDD, [0..255].
4	TCP/UDP , TCP /UDP .	`esr(config-dnat-pool)# ip port <PORT>`	<PORT> – TCP/UDP , [1..65535].
5	.	`esr(config-dnat)# ruleset <NAME>`	<NAME> – , 31 .
6	VRF, ( ).	`esr(config-dnat-ruleset)# ip vrf forwarding <VRF>`	<VRF> – VRF, 31 .
7	. , .	`esr(config-dnat-ruleset)# from { zone <NAME> | interface <IF> | tunnel <TUN> | default }`	<NAME> – ; <IF> – ; <TUN> – ; default – , .
8	c . .	`esr(config-dnat-ruleset)# rule <ORDER>`	<ORDER> – , [1..10000].
9	IP- { | }, .	`esr(config-dnat-rule)# match [not] {source|destination}-address <OBJ-GROUP-NETWORK-NAME>`	<OBJ-GROUP-NETWORK-NAME> – IP-, 31 . «any» IP- .
10	(tcp/udp-) { | }, ( ).	`esr(config-dnat-rule)# match [not] {source|destination}-port <PORT-SET-NAME>`	<PORT-SET-NAME> – , 31 . «any» TCP/UDP- .
11	IP-, ( ).	`esr(config-dnat-rule)# match [not] {protocol <TYPE> | protocol-id <ID> }`	<TYPE> – , : esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre. «any» . <ID> – IP-, [0x00-0xFF].
12	ICMP, (ICMP) ( ).	`esr(config-dnat-rule)# match [not] icmp {<ICMP_TYPE><ICMP_CODE> | <TYPE-NAME>}`	<ICMP_TYPE> – ICMP, [0..255]. <ICMP_CODE> – ICMP, [0..255]. «any» . <TYPE-NAME> – ICMP .
13	« » , , «match».	`esr(config-dnat-rule)# action destination-nat { off | pool <NAME> | netmap <ADDR/LEN> }`	off – ; pool <NAME> – , IP- / TCP/UDP ; netmap <ADDR/LEN> – IP- , . AAA.BBB.CCC.DDD/EE, AAA – DDD [0..255] EE [1..32].
14	.	`esr(config-dnat-rule)# enable`	

[1] *not,* ,

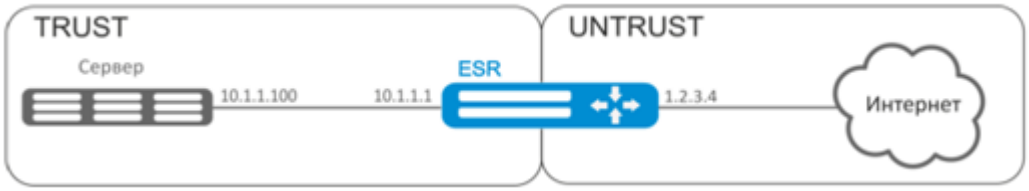
«match» «not». , .

« CLI».

Destination NAT

:

, «UNTRUST», «TRUST». – 10.1.1.100. 1.2.3.4, 80.



:

«UNTRUST» «TRUST». . IP .

```
esr# configure
esr(config)# security zone UNTRUST
esr(config-zone)# exit
esr(config)# security zone TRUST
esr(config-zone)# exit
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)#   security-zone TRUST
esr(config-if-gi)#   ip address 10.1.1.1/25
esr(config-if-gi)# exit
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip address 1.2.3.4/29
esr(config-if-te)# security-zone UNTRUST
esr(config-if-te)# exit
```

IP- , Firewall DNAT.

- NET_UPLINK – ;
- SERVER_IP – ;
- SRV_HTTP – .

```
esr(config)# object-group network NET_UPLINK
esr(config-object-group-network)# ip address 1.2.3.4
esr(config-object-group-network)# exit
```

```
esr(config)# object-group service SRV_HTTP
esr(config-object-group-service)# port 80
esr(config-object-group-service)# exit
```

```
esr(config)# object-group network SERVER_IP
esr(config-object-group-network)# ip address 10.1.1.100
esr(config-object-group-network)# exit
```

DNAT , , 1.2.3.4 .

```
esr(config)# nat destination
esr(config-dnat)# pool SERVER_POOL
esr(config-dnat-pool)# ip address 10.1.1.100
esr(config-dnat-pool)# ip port 80
esr(config-dnat-pool)# exit
```

«DNAT», . , , «UNTRUST». (match destination-address, match destination-port) . , , (action destination-nat). «enable».

```
esr(config-dnat)# ruleset DNAT
esr(config-dnat-ruleset)# from zone UNTRUST
esr(config-dnat-ruleset)# rule 1
esr(config-dnat-rule)# match destination-address NET_UPLINK
esr(config-dnat-rule)# match protocol tcp
esr(config-dnat-rule)# match destination-port SRV_HTTP
esr(config-dnat-rule)# action destination-nat pool SERVER_POOL
esr(config-dnat-rule)# enable
esr(config-dnat-rule)# exit
esr(config-dnat-ruleset)# exit
esr(config-dnat)# exit
```

, «UNTRUST» «TRUST», . , «SERVER_IP» DNAT.

```
esr(config)# security zone-pair UNTRUST TRUST
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# match destination-address SERVER_IP
esr(config-zone-pair-rule)# match destination-nat
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
esr(config)# exit
```

:

```
esr# show ip nat destination pools
esr# show ip nat destination rulesets
esr# show ip nat proxy-arp
esr# show ip nat translations
```

Source NAT

Source NAT (SNAT) , . , . . .

SNAT , . IP- .

1	.	esr(config)# nat source	
2	IP- / TCP/UDP- ( ).	esr(config-snat)# pool <NAME>	<NAME> – NAT-, 31 .
3	IP-, IP- .	esr(config-snat-pool)# ip address-range <IP>[-<ENDIP>]	<IP> – IP- , AAA.BBB.CCC.DDD, [0..255]; <ENDIP> – IP- , AAA.BBB.CCC.DDD, [0..255]. IP- , IP- IP- .
4	TCP/UDP , TCP /UDP .	esr(config-snat-pool)# ip port-range <PORT> [-<ENDPORT>]	<PORT> – TCP/UDP , [1..65535]; <ENDPORT> – TCP/UDP , [1..65535]. TCP/UDP , TCP/UDP TCP /UDP .
5	TCP/UDP , TCP /UDP .	esr(config-snat-pool)# ip port <PORT>	<PORT> – TCP/UDP , [1..65535].
6	NAT persistent.	esr(config-snat-pool)# persistent	
7	.	esr(config-snat)# ruleset <NAME>	<NAME> – , 31 .
8	VRF, ( ).	esr(config-snat-ruleset)# ip vrf forwarding <VRF>	<VRF> – VRF, 31 .

9	. , .	`esr(config-snat-ruleset)# to { zone <NAME> | interface <IF> tunnel <TUN> | | default }`	<NAME> – ; <IF> – ; <TUN> – default – , .
10	c . .	`esr(config-snat-ruleset)# rule <ORDER>`	<ORDER> – , [1..10000].
11	IP-{|}, .	`esr(config-snat-rule)# match [not] {source|destination}-address <OBJ-GROUP-NETWORK-NAME>`	<OBJ-GROUP-NETWORK-NAME> – IP-, 31 . «any» IP- .
12	IP-{|}, ( ).	`esr(config-snat-rule)# match [not] {source | destination}-port <PORT-SET-NAME>`	<PORT-SET-NAME> – , 31 . «any» TCP/UDP- .
13	IP-, ( ).	`esr(config-snat-rule)# match [not] {protocol|protocol-id} <TYPE>`	<TYPE> – , : esp, icmp, ah, eigrp, ospf, igmp, ipip, tcp, pim, udp, vrrp, rdp, l2tp, gre. «any» ; <ID> – IP-, [0x00-0xFF].
14	ICMP, ( ).	`esr(config-snat-rule)# match [not] icmp {<ICMP_TYPE><ICMP_CODE> | <TYPE-NAME>}`	<ICMP_TYPE> – ICMP, [0..255]; <ICMP_CODE> – ICMP, [0..255]. «any» ; <TYPE-NAME> – ICMP
15	« » , , «match»	`esr(config-snat-rule)# action source-nat { off | pool <NAME> | netmap <ADDR/LEN> [static] | interface [FIRST_PORT – LAST_PORT] }`	off – ; pool<NAME> – , IP- / TCP/UDP ; netmap <ADDR/LEN> – IP- , ; static – NAT. AAA.BBB.CCC.DDD/EE, AAA – DDD [0..255] EE [1..32]. interface [FIRST_PORT – LAST_PORT] – IP- . TCP/UDP-, TCP /UDP- , .
16	.	`esr(config-snat-rule)# enable`	

[1] not, ,

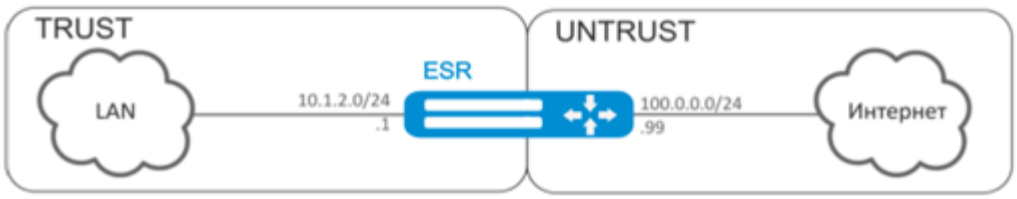
«match» «not». , .

« CLI».

1

:

10.1.2.0/24 Source NAT. SNAT 100.0.0.100-100.0.0.249.



:

, . «TRUST» «UNTRUST» .

```
esr# configure
esr(config)# security zone UNTRUST
esr(config-zone)# exit
esr(config)# security zone TRUST
esr(config-zone)# exit
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# ip address 10.1.2.1/24
esr(config-if-gi)# security-zone TRUST
esr(config-if-gi)# exit
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip address 100.0.0.99/24
esr(config-if-te)# security-zone UNTRUST
esr(config-if-te)# exit
```

SNAT «LOCAL_NET», , , «PUBLIC_POOL».

```
esr(config)# object-group network LOCAL_NET
esr(config-object-group-network)# ip address-range 10.1.2.2-10.1.2.254
esr(config-object-group-network)# exit
esr(config)# object-group network PUBLIC_POOL
esr(config-object-group-network)# ip address-range 100.0.0.100-100.0.0.249
esr(config-object-group-network)# exit
```

«TRUST» «UNTRUST» , . «LOCAL_NET» . *enable.*

```
esr(config)# security zone-pair TRUST UNTRUST
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# match source-address LOCAL_NET
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
```

SNAT. , SNAT.

```
esr(config)# nat soure
esr(config-snat)# pool TRANSLATE_ADDRESS
esr(config-snat-pool)# ip address-range 100.0.0.100-100.0.0.249
esr(config-snat-pool)# exit
```

SNAT. , , – «UNTRUST». «LOCAL_NET».

```
esr(config-snat)# ruleset SNAT
esr(config-snat-ruleset)# to zone UNTRUST
esr(config-snat-ruleset)# rule 1
esr(config-snat-rule)# match source-address LOCAL_NET
esr(config-snat-rule)# action source-nat pool TRANSLATE_ADDRESS
esr(config-snat-rule)# enable
esr(config-snat-rule)# exit
esr(config-snat-ruleset)# exit
```

ARP , , ARP Proxy. ARP Proxy , IP- «PUBLIC_POOL».

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip nat proxy-arp PUBLIC_POOL
```

, – 10.1.2.1 .

. .

```
esr(config)# ip route 0.0.0.0/0 100.0.0.1
esr(config)# exit
```

2

:

21.12.2.0/24 Source NAT (firewall). SNAT 200.10.0.100-200.10.0.249.

:

:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# ip address 21.12.2.1/24
esr(config-if-gi)# ip firewall disable
esr(config-if-gi)# exit
```

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip address 200.10.0.1/24
esr(config-if-te)# ip firewall disable
esr(config-if-te)# exit
```

SNAT «LOCAL_NET», , , «PUBLIC_POOL»:

```
esr(config)# object-group network LOCAL_NET
esr(config-object-group-network)# ip address-range 21.12.2.2-21.12.2.254
esr(config-object-group-network)# exit

esr(config)# object-group network PUBLIC_POOL
esr(config-object-group-network)# ip address-range 200.10.0.100-200.10.0.249
esr(config-object-group-network)# exit
```

SNAT.

, SNAT:

```
esr(config)# nat source
esr(config-snat)# pool TRANSLATE_ADDRESS
esr(config-snat-pool)# ip address-range 200.10.0.100-200.10.0.249
esr(config-snat-pool)# exit
```

SNAT. , , te1/0/1. «LOCAL_NET»:

```
esr(config-snat)# ruleset SNAT
esr(config-snat-ruleset)# to interface te1/0/1
esr(config-snat-ruleset)# rule 1
esr(config-snat-rule)# match source-address LOCAL_NET
esr(config-snat-rule)# action source-nat pool TRANSLATE_ADDRESS
esr(config-snat-rule)# enable
esr(config-snat-rule)# exit
esr(config-snat-ruleset)# exit
```

ARP , , ARP Proxy. ARP Proxy , IP- «PUBLIC_POOL»:

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip nat proxy-arp PUBLIC_POOL
```

, – 21.12.2.1 .

. :

```
esr(config)# ip route 0.0.0.0/0 200.10.0.254
esr(config)# exit
```

Static NAT

Static NAT — NAT . , , , --. , NAT .

Static NAT Source NAT, Source NAT, .

Static NAT

:

21.12.2.100-21.12.2.150 200.10.0.0/24. – 200.10.0.100-200.10.0.150.

:

:

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# ip address 21.12.2.1/24
esr(config-if-gi)# ip firewall disable
esr(config-if-gi)# exit
```

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip address 200.10.0.1/24
esr(config-if-te)# ip firewall disable
esr(config-if-te)# exit
```

Static NAT «LOCAL_NET», , «PUBLIC_POOL»:

```
esr(config)# object-group network LOCAL_NET
esr(config-object-group-network)# ip prefix 21.12.2.0/24
esr(config-object-group-network)# exit
```

```
esr(config)# object-group network PUBLIC_POOL
esr(config-object-group-network)# ip prefix 200.10.0.0/24
esr(config-object-group-network)# exit
```

Static NAT «PROXY»:

```
esr(config)# object-group network PROXY
esr(config-object-group-network)# ip address-range 200.10.0.100-200.10.0.150
esr(config-object-group-network)# exit
```

Static NAT SNAT. , , te1/0/1. «LOCAL_NET» «PUBLIC_POOL».

```
esr(config)# nat source
esr(config-snat)# ruleset SNAT
esr(config-snat-ruleset)# to interface te1/0/1
esr(config-snat-ruleset)# rule 1
esr(config-snat-rule)# match source-address LOCAL_NET
esr(config-snat-rule)# match destination-address PUBLIC_POOL
esr(config-snat-rule)# action source-nat netmap 200.10.0.0/24 static
esr(config-snat-rule)# enable
esr(config-snat-rule)# exit
esr(config-snat-ruleset)# exit
```

ARP , «PROXY», ARP Proxy. ARP Proxy , IP- «PROXY».

```
esr(config)# interface tengigabitethernet 1/0/1
esr(config-if-te)# ip nat proxy-arp PROXY
```

c 200.10.0.0/24, – 21.12.2.1 .

.

```
esr# commit
Configuration has been successfully committed
esr# confirm
Configuration has been successfully confirmed
```

:

```
esr# show ip nat translations
```

HTTP/HTTPS-

1	C URL	esr(config)# object-group url <NAME>	
2		esr(config-object-group-url)# url <URL>	<URL> – , .
3		esr(config)# ip http profile <NAME>	<NAME> – .
4		esr(config-profile)# default action {deny|permit|redirect} [redirect-url <URL>]	<URL> – , .
5	()	esr(config-profile)# description <description>	<description> – 255 .
6	URL (/ /)()	esr(config-profile)# urls {local|remote} <URL_OBJ_GROUP_NAME> action {deny|permit|redirect} [redirect-url <URL>]	<URL_OBJ_GROUP_NAME> – , URL.
7	, URL()	esr(config)# ip http proxy server-url <URL>	<URL> – , url.
8	()	esr(config)# ip http proxy listen-ports <OBJ_GROUP_NAME>	<OBJ_GROUP_NAME> – , 31 .

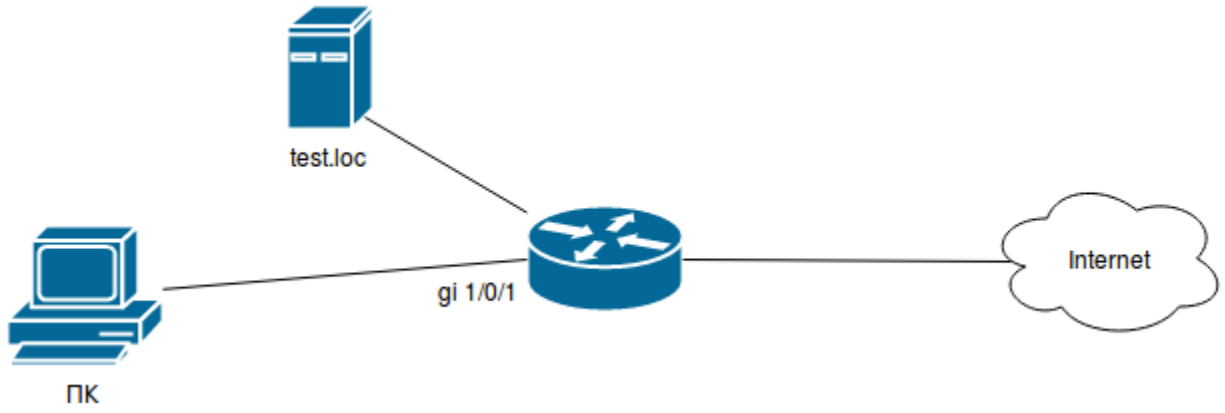
9	( )	esr(config)# ip https proxy listen-ports <OBJ_GROUP_NAME>	<OBJ_GROUP_NAME> – , 31 .
10	( )	esr(config)# ip https proxy redirect-port <PORT>	<PORT> – , [1..65535]. 3128
11	HTTP-	esr(config-if)# ip http proxy <PROFILE_NAME>	<PROFILE_NAME> –
12	HTTPS-	esr(config-if)# ip https proxy <PROFILE_NAME>	<PROFILE_NAME> –
13	, .	esr(config)# object-group service <obj-group-name>	<obj-group-name> – , 31 .
14	( ).	esr(config-object-group-service)# description <description>	<description> – , 255 .
15	(tcp/udp) .	esr(config-object-group-service)# port-range 3128-3135	- ESR 10 http proxy + cpu ESR - 1 https proxy + cpu ESR + cpu ESR * 2 - 1
16	.	esr(config)# security zone-pair <src-zone-name1> self	<src-zone-name> – , ip http proxy ip https proxy. self – , ESR.
17	.	esr(config-zone-pair)# rule <rule-number>	<rule-number> – 1..10000.
18	( ).	esr(config-zone-rule)# description <description>	<description> – 255 .
19	.	esr(config-zone-rule)# action <action> [log]	<action> – permit log – , .
20	IP-,	esr(config-zone-rule)# match protocol <protocol-type>	<protocol-type> – tcp - ESR ESR.
21	TCP/UDP- , ( ).	esr(config-zone-rule)# match [not] destination-port <obj-group-name>	<obj-group-name> – , 12
22	.	esr(config-zone-rule)# enable	

Firewall ESR , Self.

HTTP-

:

URL .



:

URL, . - URL:

```
esr# configure
esr(config)# object-group url test1
esr(config-object-group-url)# url http://speedtest.net/
esr(config-object-group-url)# url http://www.speedtest.net/
esr(config-object-group-url)# url https://speedtest.net/
esr(config-object-group-url)# url https://www.speedtest.net/
esr(config-object-group-url)# exit
```

:

```
esr(config)# ip http profile list1
esr(config-profile)# default action permit
esr(config-profile)# urls local test1 action redirect redirect-url http://test.loc
esr(config-profile)# exit
```

‘list’:

```
esr(config)# interface gi 1/0/1
esr(config-if)# ip http proxy list1
esr(config-if)# ip https proxy list1
```

Firewall, :

ESR-20 4 CPU.

http proxy 3128 3131

https proxy 3132 3135

-:

```
esr(config)# object-group service proxy
esr(config-object-group-service)# port-range 3128-3135
esr(config-object-group-service)# exit
```

:

```
esr(config)# security zone-pair LAN self
esr(config-zone-pair)# rule 50
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol tcp
esr(config-zone-pair-rule)# match destination-port proxy
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
```

NTP

NTP (. Network Time Protocol —) — IP , UDP, .

1	NTP.	esr(config)# ntp enable	
2	IP NTP , NTP .	esr(config)# ntp { server | peer } { <IP> }	<IP> – IP- (), AAA.BBB.CCC.DDD, [0..255].
3	( ).	esr(config-ntp)# key <ID>	<ID> – , [1..255].
4	NTP- ( ).	esr(config-ntp)# maxpoll <INTERVAL>	<INTERVAL> – . , , , [10..17]. : 10 (2^{10}= 1024 17 4).

5	NTP- ( ).	esr(config-ntp)# minpoll <INTERVAL>	<INTERVAL> – , , [4..6]. : 6 (2^6= 64 1 4).
6	NTP- ( ).	esr(config-ntp)# prefer	
7	IP-, ntp- ( ).	esr(config)# ntp access-addresses <NAME>	<NAME> – IP-, 31 .
8	( ).	esr(config)# ntp authentication trusted-key <ID>	<ID> – .
9	( ).	esr(config)# ntp authentication key-chain <WORD>	<WORD> - .
10	NTP ( ).	esr(config)# ntp authentication enable	
11	NTP- VRF ( ).	esr(config)# ntp broadcast-client enable	
12	DSCP IP- NTP- ( ).	esr(config)# ntp dscp <DSCP>	<DSCP> – DSCP, [0..63] : 46
13	query-only, NTP IP- ( ).	esr(config)# ntp object-group query-only <NAME>	<NAME> – IP-, 31 .
14	serve-only, NTP IP- ( ).	esr(config)# ntp object-group serve-only <NAME>	<NAME> – IP-, 31 .
15	source-IP- NTP-peer ( ).	esr(config)# ntp source address <ADDR>	<ADDR> – IP-, AAA.BBB.CCC.DDD, [0..255].
16	( ).	esr# set date <TIME> [<DAY> <MONTH> [<YEAR>]]	<TIME> – , HH:MM:SS, : HH – , [0..23]; MM – , [0 .. 59]; SS – , [0 .. 59]. <DAY> – , [1..31]; <MONTH> – , [January/February/March/April/May/June/July/August/September/October /November/December]; <YEAR> – , [2001..2037].

:

NTP .

IP- esr - 192.168.52.8,

IP- NTP – 192.168.52.41.

:

:

- gi1/0/1;
- IP- gi1/0/1, IP- NTP-.

:

```
security zone untrust
exit
object-group service NTP
  port-range 123
exit
interface gigabitethernet 1/0/1
  security-zone untrust
  ip address 192.168.52.8/24
exit
security zone-pair untrust self
  rule 10
    action permit
    match protocol udp
    match destination-port NTP
    enable
  exit
exit
```

:

:

```
esr(config)# ntp enable
```

NTP-:

```
esr-(config)# ntp server 192.168.52.41
```

NTP- ():

```
esr-1000(config-ntp)# prefer
```

NTP-:

```
esr(config-ntp)# minpoll 4
esr(config-ntp)# end
esr# commit
esr# confirm
```

NTP:

```
esr# show ntp configuration
```

NTP- ():

```
esr# show ntp peers
```