

BRAS (Broadband Remote Access Server)

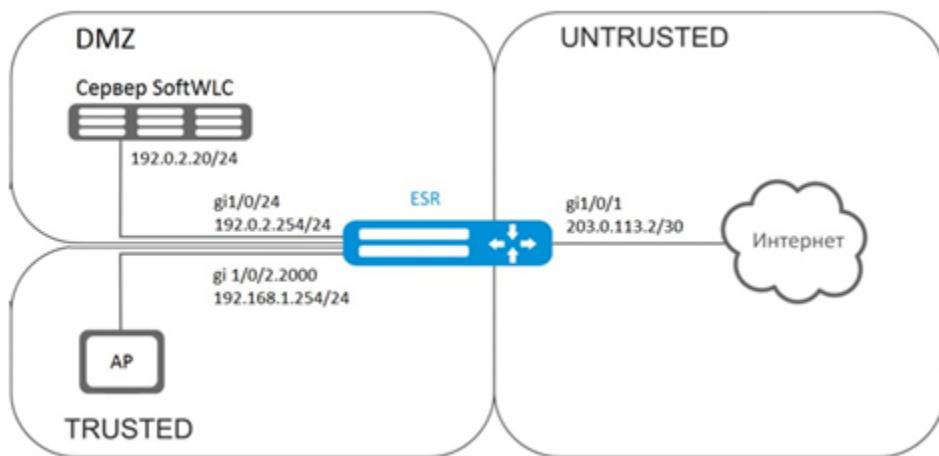
- SoftWLC
- SoftWLC

1	RADIUS- .	esr(config)# radius -server host { <IP-ADDR> <IPV6-ADDR> } [vrf <VRF>]esr(config-radius-server)#	<IP-ADDR> – IP- RADIUS-, AAA.BBB.CCC.DDD, [0..255]; <IPV6-ADDR> – IPv6- RADIUS-, X:X:X::X, [0..FFFF] <VRF> – VRF, 31.
2	RADIUS-.	esr(config-radius-server)# key ascii-text { <TEXT> encrypted <ENCRYPTED-TEXT> }	<TEXT> – [8..16] ASCII; <ENCRYPTED-TEXT> – , [8..16], [16..32].
3	.	esr(config)# aaa radius-profile <NAME>	<NAME> – , 31.
4	AAA RADIUS-.	esr(config-aaa-radius-profile)# radius-server host { <IP-ADDR> <IPV6-ADDR> }	<IP-ADDR> – IP- RADIUS-, AAA.BBB.CCC.DDD, [0..255]; <IPV6-ADDR> – IPv6- RADIUS-, X:X:X::X, [0..FFFF].
5	DAS-.	esr(config)# das-server <NAME>	<NAME> – DAS-, 31.
6	DAS-.	esr(config-das-server)# key ascii-text {<TEXT> encrypted <ENCRYPTED-TEXT> }	<TEXT> – [8..16] ASCII; <ENCRYPTED-TEXT> – , [8..16], [16..32].
7	AAA DAS-.	esr(config)# aaa das-profile <NAME>	<NAME> – DAS-, 31.
8	DAS- DAS-.	esr(config-aaa-das-profile)# das-server <NAME>	<NAME> – DAS-, 31.
9	BRAS.	esr(config)# subscriber-control [vrf <VRF>]	<VRF> – VRF, 31, .
10	(DAS), CoA- PCRF	esr(config-subscriber-control)# aaa das-profile <NAME>	<NAME> – (DAS), 31.
11	RADIUS-	esr(config-subscriber-control)# aaa services-radius-profile <NAME>	<NAME> – RADIUS-, 31.
12	RADIUS- .	esr(config-subscriber-control)# aaa sessions-radius-profile <NAME>	<NAME> – RADIUS-, 31.
13	IP-, IP- RADIUS .	esr(config-subscriber-control)# nas-ip-address <ADDR>	<ADDR> – IP-, AAA.BBB.CCC.DDD, [0..255].
14	MAC- () .	esr(config-subscriber-control)# session mac-authentication	
15	(DHCP, DNS ..) .	esr(config-subscriber-control)# bypass-traffic-all <NAME>	<NAME> – ACL, 31.
16	.	esr(config-subscriber-control)# default-service	
17	QoS- .	esr(config-subscriber-default-service)# class-map <NAME>	<NAME> – , 31.

18	URL, HTTP/HTTPS .	esr(config-subscriber-default-service)# filter-name { local<LOCAL-NAME> remote<REMOTE-NAME> }	<LOCAL-NAME> – URL, 31 ; <REMOTE-NAME> – URL , 31.
19	, HTTP/HTTPS-, URL URL, «filter-name».	esr(config-subscriber-default-service)# filter-action<ACT>	<ACT> – : • permit – ; • deny – ; redirect <URL> – URL, 255 .
20	, HTTP/HTTPS-, URL URL, «filter-name».	esr(config-subscriber-default-service)# default - action<ACT>	<ACT> – : • permit – ; • deny – ; redirect <URL> – URL, 255 .
21	.	esr(config-subscriber-control)# enable	
22	(,-) (.)	esr(config-if)# location <ID>	<ID> – , 220 .
23	.	esr(config-if-gi)# service-subscriber-control {any object-group <NAME>}	<NAME> – IP-, 31 .
24	(.).	esr(config-subscriber-control)# quota-expired-reauth	
25	IP-. (.)	esr(config-subscriber-control)# session ip-authentication	
26	backup BRAS (.).	esr(config-subscriber-control)# backup traffic-processing transparent	
27	, URL (.).	esr(config)# subscriber-control unused-filters-remove-delay <DELAY>	<DELAY> – , [10800..86400].
28	, , (.).	esr(config-subscriber-default-service)# session-timeout <SEC>	<SEC> – , [120..3600].
29	VRRP-, (.).	esr(config-subscriber-control)# vrrp-group <GRID>	<GRID> – VRRP-, [1..32].
30	TCP- HTTP Proxy- (.).	esr(config-subscriber-control)# ip proxy http listen-ports <NAME>	<NAME> – TCP/UDP-, 31 .
31	HTTP Proxy- (.).	esr(config-subscriber-control)# ip proxy http redirect-port <PORT>	<PORT> – , [1..65535].
32	TCP- HTTPS Proxy- (.).	esr(config-subscriber-control)# ip proxy https listen-ports <NAME>	<NAME> – TCP/UDP-, 31 .
33	HTTPS Proxy- (.).	esr(config-subscriber-control)# ip proxy https redirect-port <PORT>	<PORT> – , [1..65535].
34	IP-, IP- Proxy- HTTP/HTTPS (.).	esr(config-subscriber-control)# ip proxy source-address <ADDR>	<ADDR> – IP- , AAA.BBB.CCC.DDD, [0..255].
35	URL- , (.)	esr(config)# subscriber-control apps-server-url <URL>	<URL> – , 8 255 .
36	()	esr(config-if-gi)# subscriber-control application-filter <NAME>	<NAME> – , 31 .
37	/ BRAS()	esr(config-subscriber-control)# thresholds sessions-number high <Threshold>	<Threshold> – BRAS, [0-50000] – ESR-1700 • [0-10000] – ESR-1200/1000 • [0-1000] – ESR-100/200

38	/ BRAS()	<pre>esr(config-subscriber-control)# thresholds sessions-number low <Threshold></pre>	<pre><Threshold> - BRAS, [0-50000] - ESR-1700 • [0-10000] - ESR-1200/1000 • [0-1000] - ESR-100/200</pre>
----	----------	---	--

SoftWLC



SoftWLC. SoftWLC :

[v1.19_SoftWLC](#) – SoftWLC;

[v1.19_](#) – SoftWLC .

BRAS,

, :

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

```
esr(config)# interface gigabitethernet 1/0/1
esr(config-if-gi)# security-zone untrusted
esr(config-if-gi)# ip address 203.0.113.2/30
esr(config-if-gi)# service-policy dynamic upstream
esr(config-if-gi)# exit
esr(config)# ip route 0.0.0.0/0 203.0.113.1
```

SoftWLC:

```
esr (config)# interface gigabitethernet 1/0/24
esr (config-if-gi)# security-zone dmz
esr (config-if-gi)# ip address 192.0.2.1/24
esr (config-if-gi)# exit
```

Wi-Fi .

```
esr(config)# bridge 2
esr(config-bridge)# security-zone trusted
esr(config-bridge)# ip address 192.168.0.254/24
esr(config-bridge)# ip helper-address 192.0.2.20
esr(config-bridge)# service-subscriber-control object-group users
esr(config-bridge)# location ssid1
esr(config-bridge)# enable
esr(config-bridge)# exit
esr(config)# interface gigabitethernet 1/0/2.2000
esr(config-subif)# bridge-group 1
esr(config-subif)# exit
esr(config)# interface gigabitethernet 1/0/2
esr(config-if-gi)# service-policy dynamic downstream
esr (config-if-gi)# exit
```



, location (bridge 2) .

, -, eltex-radius IP- SoftWLC. - SoftWLC.

:

```
esr(config)# radius-server host 192.0.2.20
esr(config-radius-server)# key ascii-text password
esr(config-radius-server)# auth-port 31812
esr (config-radius-server)# acct-port 31813
esr (config-radius-server)# exit
```

AAA:

```
esr(config)# aaa radius-profile RADIUS
esr(config-aaa-radius-profile)# radius-server host 192.0.2.20
esr(config-aaa-radius-profile)# exit
```

DAS (Direct-attached storage) :

```
esr(config)# object-group network server
esr(config-object-group-network)# ip address-range 192.0.2.20
esr(config-object-group-network)# exit
esr(config)# das-server CoA
esr(config-das-server)# key ascii-text password
esr(config-das-server)# port 3799
esr(config-das-server)# clients object-group server
esr(config-das-server)# exit
esr(config)# aaa das-profile CoA
esr(config-aaa-das-profile)# das-server CoA
esr(config-aaa-das-profile)# exit
```

trusted, DHCP DNS . DHCP DNS :

```
esr(config)# ip access-list extended DHCP
esr(config-acl)# rule 10
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol udp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port 68
esr(config-acl-rule)# match destination-port 67
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# rule 11
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol udp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 53
esr(config-acl-rule)# enable
esr(config-acl-rule)#exit
esr(config-acl)# exit
```

:

```
esr(config)# ip access-list extended WELCOME
esr(config-acl)# rule 10
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol any
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# exit
esr (config)# ip access-list extended INTERNET
esr(config-acl)# rule 10
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol any
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# exit
```

web- :

```
esr(config)# object-group url defaultservice
esr(config-object-group-url)# url http://eltex.nsk.ru
esr(config-object-group-url)# exit
```

URL SoftWLC(IP- SoftWLC, , URL):

```
esr(config)# subscriber-control filters-server-url http://192.0.2.20:7070/Filters/file/
```

BRAS, NAS IP SoftWLC, – IP- gigabitethernet 1/0/24:

```
esr(config)# subscriber-control
esr(config-subscriber-control)# aaa das-profile CoA
esr(config-subscriber-control)# aaa sessions-radius-profile RADIUS
esr(config-subscriber-control)# nas-ip-address 192.0.2.1
esr(config-subscriber-control)# session mac-authentication
esr(config-subscriber-control)# bypass-traffic-acl DHCP
esr(config-subscriber-control)# default-service
esr(config-subscriber-default-service)# class-map INTERNET
esr(config-subscriber-default-service)# filter-name local defaultservice
esr(config-subscriber-default-service)# filter-action permit
esr(config-subscriber-default-service)# default-action redirect http://192.0.2.20:8080/eltex_portal/
esr(config-subscriber-default-service)# session-timeout 3600
esr(config-subscriber-default-service)# exit
esr(config-subscriber-control)# enable
esr(config-subscriber-control)# exit
```

```
esr(config)# object-group service telnet
esr(config-object-group-service)# port-range 23
esr(config-object-group-service)# exit
esr(config)# object-group service ssh
esr(config-object-group-service)# port-range 22
esr(config-object-group-service)# exit
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)# object-group service ntp
esr(config-object-group-service)# port-range 123
esr(config-object-group-service)# exit
```

trusted dmz:

```
esr(config)# security zone-pair trusted untrusted
esr(config-zone-pair)# rule 10
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol any
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
esr(config)# security zone-pair dmz untrusted
esr(config-zone-pair)# rule 10
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol any
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
esr(config)# security zone-pair dmz trusted
esr(config-zone-pair)# rule 10
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol any
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit
```

DHCP trusted dmz:

```

esr (config)# security zone-pair trusted dmz
esr (config-zone-pair)# rule 10
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol udp
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# match source-port dhcp_client
esr(config-zone-pair-rule)# match destination-port dhcp_server
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# exit

```

ICMP , BRAS -- TCP 3129/3128 (NetPort Discovery Port/Active API Server Port):

```

esr(config)# object-group service bras
esr(config-object-group-service)# port-range 3129
esr(config-object-group-service)# port-range 3128
esr(config-object-group-service)# exit
esr(config)# security zone-pair trusted self
esr(config-zone-pair)# rule 10
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol tcp
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# match source-port any
esr(config-zone-pair-rule)# match destination-port bras
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair)# rule 20
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol icmp
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair-rule)# exit
esr(config)# security zone-pair dmz self
esr(config-zone-pair)# rule 20
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol icmp
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair-rule)# exit
esr(config)# security zone-pair untrusted self
esr(config-zone-pair)# rule 20
esr(config-zone-pair-rule)# action permit
esr(config-zone-pair-rule)# match protocol icmp
esr(config-zone-pair-rule)# match source-address any
esr(config-zone-pair-rule)# match destination-address any
esr(config-zone-pair-rule)# enable
esr(config-zone-pair-rule)# exit
esr(config-zone-pair-rule)# exit

```

DHCP-Relay:

```

esr(config)# ip dhcp-relay

```

SNAT gigabitethernet 1/0/1:

```
esr(config)# nat source
esr(config-snat)# ruleset inet
esr(config-snat-ruleset)# to interface gigabitethernet 1/0/1
esr(config-snat-ruleset)# rule 10
esr(config-snat-rule)# match source-address any
esr(config-snat-rule)# action source-nat interface
esr(config-snat-rule)# enable
esr(config-snat-rule)# end
```

SoftWLC

```
:
BRAS SoftWLC.
:
10.10.0.0/16,  FreeRADIUS- 192.168.1.1/24
:
1:
RADIUS-.
FreeRADIUS- , . users  FreeRADIUS :
:
<MACADDR> Cleartext-Password := <MACADDR>
#
User-Name = <USER_NAME>,
#
Session-Timeout = <SECONDS>,
#
Idle-Timeout = <SECONDS>,
#
Acct-Interim-Interval = <SECONDS>,
#
# (A- ,N- )
Cisco-Account-Info = "{A|N}<SERVICE_NAME>"
:
<SERVICE_NAME> Cleartext-Password := <MACADDR>
#
# class-map ESR
Cisco-AVPair = "subscriber:traffic-class=<CLASS_MAP>" ,
```

```
# , ESR (permit, deny, redirect)
```

```
Cisco-AVPair = "subscriber:filter-default-action=<ACTION>>,
```

```
# IP (enabled-uplink, enabled-downlink, enabled, disabled)
```

```
Cisco-AVPair = "subscriber:flow-status=<STATUS>"
```

```
clients.conf , ESR:
```

```
client ESR {  
ipaddr = <SUBNET>  
secret = <RADIUS_KEY>  
}
```

```
RADIUS- :
```

```
«clients.conf» :
```

```
client BRAS {  
ipaddr = 192.168.1.1  
secret = password  
}
```

```
«users» (<MAC> MAC):
```

```
"54-E1-AD-8F-37-35" Cleartext-Password := "54-E1-AD-8F-37-35"  
User-Name = "Bras_user",  
Session-Timeout = 259200,  
Idle-Timeout = 259200,  
Cisco-AVPair += "subscriber:policer-rate-in=1000",  
Cisco-AVPair += "subscriber:policer-rate-out=1000",  
Cisco-AVPair += "subscriber:policer-burst-in=188",  
Cisco-AVPair += "subscriber:policer-burst-out=188",  
Cisco-Account-Info = "AINTERNET"  
INTERNET Cleartext-Password := "INTERNET"  
User-Name = "INTERNET",  
Cisco-AVPair = "subscriber:traffic-class=INTERNET",  
Cisco-AVPair += "subscriber:filter-default-action=permit"
```

```
2:
```

```
ESR.
```

```
BRAS BRAS.
```

```
esr(config)# do sh licence  
Licence information  
-----  
Name: Eltex  
Version: 1.0  
Type: ESR-X  
S/N: NP00000000  
MAC: XX:XX:XX:XX:XX:XX  
Features:  
    BRAS - Broadband Remote Access Server
```

```
RADIUS-:
```

```
esr(config)# radius-server host 192.168.1.2
esr(config-radius-server)# key ascii-text encrypted 8CB5107EA7005AFF
esr(config-radius-server)# source-address 192.168.1.1
esr(config-radius-server)# exit
```

AAA:

```
esr(config)# aaa radius-profile bras_radius
esr(config-aaa-radius-profile)# radius-server host 192.168.1.2
esr(config-aaa-radius-profile)# exit
esr(config)# aaa radius-profile bras_radius_servers
esr(config-aaa-radius-profile)# radius-server host 192.168.1.2
esr(config-aaa-radius-profile)# exit
```

DAS-:

```
esr(config)# das-server das
esr(config-das-server)# key ascii-text encrypted 8CB5107EA7005AFF
esr(config-das-server)# exit
esr(config)# aaa das-profile bras_das
esr(config-aaa-das-profile)# das-server das
esr(config-aaa-das-profile)# exit
esr(config)# vlan 10
esr(config-vlan)# exit
```

:

```

esr(config)# ip access-list extended BYPASS
esr(config-acl)# rule 1
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol udp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port 68
esr(config-acl-rule)# match destination-port 67
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# rule 2
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol udp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 53
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config)# ip access-list extended INTERNET
esr(config-acl)# rule 1
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol any
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config)# ip access-list extended WELCOME
esr(config-acl)# rule 10
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol tcp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 443
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# rule 20
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol tcp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 8443

```

```

esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# rule 30
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol tcp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 80
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit
esr(config-acl)# rule 40
esr(config-acl-rule)# action permit
esr(config-acl-rule)# match protocol tcp
esr(config-acl-rule)# match source-address any
esr(config-acl-rule)# match destination-address any
esr(config-acl-rule)# match source-port any
esr(config-acl-rule)# match destination-port 8080
esr(config-acl-rule)# enable
esr(config-acl-rule)# exit

```

URL , , http-proxy BRAS :

```
esr(config)# object-group url defaultserv
esr(config-object-group-url)# url http://eltex.nsk.ru
esr(config-object-group-url)# url http://ya.ru
esr(config-object-group-url)# url https://ya.ru
esr(config-object-group-url)# exit
```

BRAS, NAS IP RADIUS- – IP- gigabitethernet 1/0/2:

```
esr(config)# subscriber-control
esr(config-subscriber-control)# aaa das-profile bras_das
esr(config-subscriber-control)# aaa sessions-radius-profile bras_radius
esr(config-subscriber-control)# aaa services-radius-profile bras_radius_servers
esr(config-subscriber-control)# nas-ip-address 192.168.1.1
esr(config-subscriber-control)# session mac-authentication
esr(config-subscriber-control)# bypass-traffic-acl BYPASS
esr(config-subscriber-control)# default-service
esr(config-subscriber-default-service)# class-map BYPASS
esr(config-subscriber-default-service)# filter-name local defaultserv
esr(config-subscriber-default-service)# filter-action permit
esr(config-subscriber-default-service)# default-action redirect http://192.
168.1.2:8080/eltex_portal
esr(config-subscriber-default-service)# session-timeout 121
esr(config-subscriber-default-service)# exit
esr(config-subscriber-control)# enable
esr(config-subscriber-control)# exit
```

, BRAS ():

```
esr(config)# bridge 10
esr(config-bridge)# vlan 10
esr(config-bridge)# ip firewall disable
esr(config-bridge)# ip address 10.10.0.1/16
esr(config-bridge)# ip helper-address 192.168.1.2
esr(config-bridge)# service-subscriber-control any
esr(config-bridge)# location USER
esr(config-bridge)# protected-ports
esr(config-bridge)# protected-ports exclude vlan
esr(config-bridge)# enable
esr(config-bridge)# exit
```

RADIUS-:

```
esr(config)# interface gigabitethernet 1/0/2
esr(config-if-gi)# ip firewall disable
esr(config-if-gi)# ip address 192.168.1.1/24
esr(config-if-gi)# exit
```

:

```
esr(config)# interface gigabitethernet 1/0/3.10
esr(config-subif)# bridge-group 10
esr(config-subif)# ip firewall disable
esr(config-subif)# exit
```

SNAT gigabitethernet 1/0/2:

```
esr(config)# nat source
esr(config-snat)# ruleset factory
esr(config-snat-ruleset)# to interface gigabitethernet 1/0/2
esr(config-snat-ruleset)# rule 10
esr(config-snat-rule)# description "replace 'source ip' by outgoing interface ip address"
esr(config-snat-rule)# match protocol any
esr(config-snat-rule)# match source-address any
esr(config-snat-rule)# match destination-address any
esr(config-snat-rule)# action source-nat interface
esr(config-snat-rule)# enable
esr(config-snat-rule)# exit
esr(config-snat-ruleset)# exit
esr(config-snat)# exit
esr(config)# ip route 0.0.0.0/0 192.168.1.2
```

:

```
esr(config) # do commit
esr(config) # do confirm
```

- :

```
esr # sh subscriber-control sessions status

Session id      User name      IP address      MAC address      Interface      Domain
-----  -----  -----  -----  -----
1729382256910270473    Bras_user    10.10.0.3  54:e1:ad:8f:37:35    g1/0/3.10      --
```