

- Netflow
  -
- sFlow
  -
- SNMP
  -
- Zabbix-agent/proxy
  - zabbix-agent
  - zabbix-server
- Syslog
  -
- -
- -

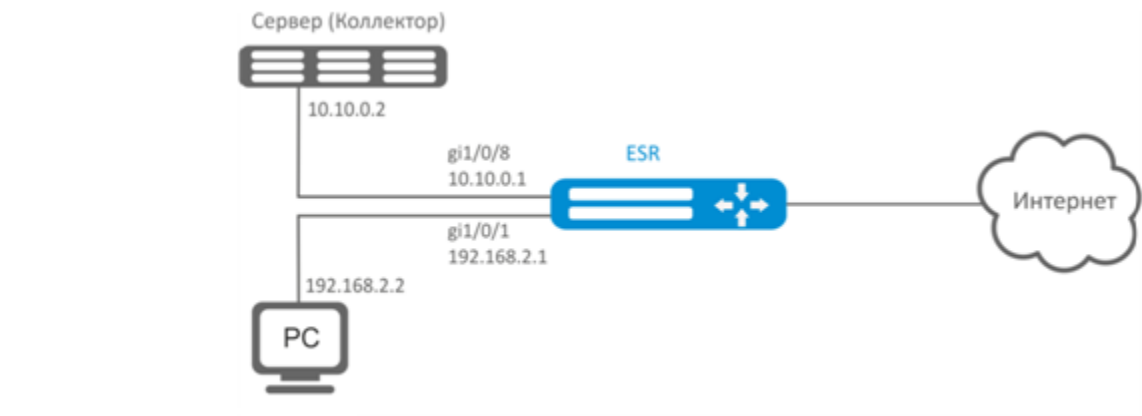
Netflow

Netflow — , . Netflow ( , , . ) () . .

1	Netflow-.	<b>esr(config)# netflow version &lt;VERSION&gt;</b>	<VERSION> – Netflow-: 5,9 10.
2	.	<b>esr(config)# netflow max-flows &lt;COUNT&gt;</b>	<COUNT> – , [10000..2000000]. : 512000.
3	, .	<b>esr(config)# netflow inactive-timeout &lt;TIMEOUT&gt;</b>	<TIMEOUT> – , , [0..240]. : 15 .
4	Netflow-.	<b>esr(config)# netflow refresh-rate &lt;RATE&gt;</b>	<RATE> – , , [1..10000]. : 10.
5	Netflow .	<b>esr(config)# netflow enable</b>	
6	Netflow .	<b>esr(config)# netflow collector &lt;ADDR&gt;</b>	<ADDR> – IP-, AAA.BBB.CCC.DDD, [0..255].
7	Netflow- .	<b>esr(config-netflow-host)# port &lt;PORT&gt;</b>	<PORT> – UDP-, [1..65535]. : 2055.
8	Netflow- // .	<b>esr(config-if-gi)# ip netflow export</b>	

:

gi1/0/1 gi1/0/8 .



- gi1/0/1, gi1/0/8 firewall «ip firewall disable».
- IP- .

IP- :

```
esr(config)# netflow collector 10.10.0.2
```

```
netflow gi1/0/1:  
  
esr(config)# interface gigabitethernet 1/0/1  
esr(config-if-gi)# ip netflow export
```

netflow :

```
sr(config)# netflow enable
```

Netflow :

```
esr# show netflow statistics
```

Netflow sFlow, sFlow.

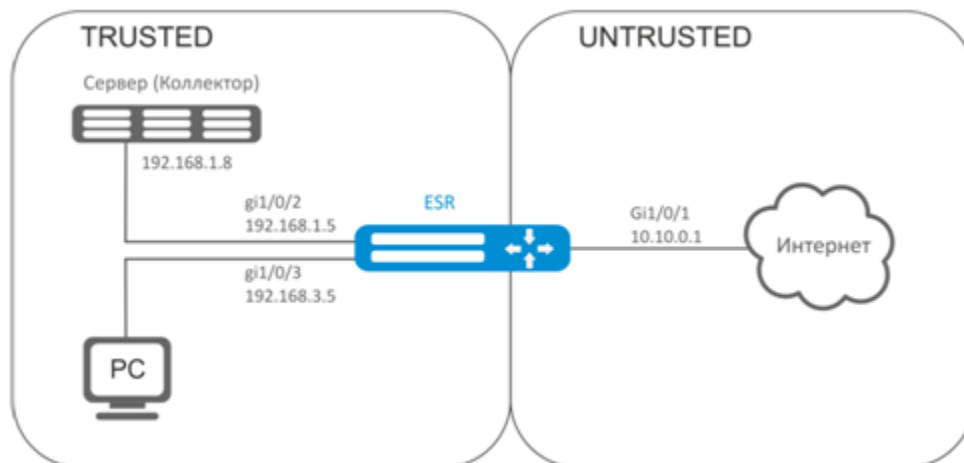
## sFlow

Sflow — , , .

1	sFlow-	<b>esr(config)# sflow sampling-rate &lt;RATE&gt;</b>	<RATE> – , [1..10000000]. 10 . : 1000.
2	,	<b>esr(config)# sflow poll-interval &lt;TIMEOUT&gt;</b>	<TIMEOUT> – , , [1..10000]. : 10 .
3	sFlow .	<b>esr(config)# sflow enable</b>	
4	sFlow .	<b>esr(config)# sflow collector &lt;ADDR&gt;</b>	<ADDR> – IP- , AAA.BBB.CCC.DDD, [0..255].
5	sFlow- // .	<b>esr(config-if-gi)# ip sflow export</b>	

:

trusted untrusted.



:

ESR :

```
esr# configure
esr(config)# security zone TRUSTED
esr(config-zone)# exit
esr(config)# security zone UNTRUSTED
esr(config-zone)# exit
```

:

```
esr(config)# interface gi1/0/1
esr(config-if-gi)# security-zone UNTRUSTED
esr(config-if-gi)# ip address 10.10.0.1/24
esr(config-if-gi)# exit
esr(config)# interface gi1/0/2-3
esr(config-if-gi)# security-zone TRUSTED
esr(config-if-gi)# exit
esr(config)# interface gi1/0/2
esr(config-if-gi)# ip address 192.168.1.5/24
esr(config-if-gi)# exit
esr(config)# interface gi1/0/3
esr(config-if-gi)# ip address 192.168.3.5/24
esr(config-if-gi)# exit
```

IP- :

```
esr(config)# sflow collector 192.168.1.8
```

sFlow «rule1» TRUSTED-UNTRUSTED:

```
esr(config)# security zone-pair TRUSTED UNTRUSTED
esr(config-zone-pair)# rule 1
esr(config-zone-pair-rule)# action sflow-sample
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
```

sFlow :

```
sr(config)# sflow enable
```

sFlow      [Netflow](#).

## SNMP

SNMP (. *Simple Network Management Protocol* — ) — , IP- TCP/UDP. SNMP , .

1	SNMP-	<b>esr(config)# snmp-server</b>	
2	community SNMPv2c.	<b>esr(config)# snmp-server community</b> <b>&lt;COMMUNITY&gt; [ &lt;TYPE&gt; ]</b> <b>[ { &lt;IP-ADDR&gt;   &lt;IPv6-ADDR&gt; } ]</b> <b>[ client-list &lt;OBJ-GROUP-NETWORK-</b> <b>NAME&gt; ]</b> <b>[ &lt;VERSION&gt; ] [ view &lt;VIEW-NAME&gt;</b> <b>] [ vrf &lt;VRF&gt; ]</b>	<b>&lt;COMMUNITY&gt; –</b> SNMP;  <b>&lt;TYPE&gt; –</b> : <ul style="list-style-type: none"><li>• <b>ro</b> – ;</li><li>• <b>rw</b> – .</li></ul> <b>&lt;IP-ADDR&gt; –</b> IP- , , AAA.BBB.CCC.DDD, [0..255]; <b>&lt;IPv6-ADDR&gt; –</b> IPv6- , X:X:X:X::X, [0..FFFF]; <b>&lt;OBJ-GROUP-NETWORK-NAME&gt; –</b> IP-, snmp-, 31 ; <b>&lt;VERSION&gt; –</b> snmp, community, v1 v2c; <b>&lt;VIEW-NAME&gt; –</b> SNMP view, 31 ; <b>&lt;VRF&gt; –</b> VRF, , 31 .
3	SNMP,	<b>esr(config)# snmp-server contact</b> <b>&lt;CONTACT&gt;</b>	<b>&lt;CONTACT&gt; –</b> , 255 .
4	DSCP IP- SNMP- ( ).	<b>esr(config)# snmp-server dscp</b> <b>&lt;DSCP&gt;</b>	<b>&lt;DSCP&gt; –</b> DSCP, [0..63].  : 63.
5	snmp- ( )	<b>esr(config)# snmp-server system-</b> <b>shutdown</b>	
6	C SNMPv3-.	<b>esr(config)# snmp-server user</b> <b>&lt;NAME&gt;</b>	<b>&lt;NAME&gt; –</b> , 31 .
7	SNMP,	<b>esr(config)# snmp-server location</b> <b>&lt;LOCATION&gt;</b>	<b>&lt;LOCATION&gt; –</b> , 255 .
8	SNMPv3.	<b>esr(config-snmp-user)# access</b> <b>&lt;TYPE&gt;</b>	<b>&lt;TYPE&gt; –</b> : <ul style="list-style-type: none"><li>• <b>ro</b> – ;</li><li>• <b>rw</b> – .</li></ul>
9	SNMPv3.	<b>esr(config-snmp-user)#</b> <b>authentication access &lt;TYPE&gt;</b>	<b>&lt;TYPE&gt; –</b> : <ul style="list-style-type: none"><li>• <b>auth</b> – ;</li><li>• <b>priv</b> – .</li></ul>
10	SNMPv3-.	<b>esr(config-snmp-user)#</b> <b>authentication algorithm</b> <b>&lt;ALGORITHM&gt;</b>	<b>&lt;ALGORITHM&gt; –</b> : <ul style="list-style-type: none"><li>• <b>md 5</b> – md5;</li><li>• <b>sha 1</b> – sha1.</li></ul>
11	SNMPv3-.	<b>esr(config-snmp-user)#</b> <b>authentication key ascii-text</b> <b>{ &lt;CLEAR-TEXT&gt;   encrypted</b> <b>&lt;ENCRYPTED-TEXT&gt; }</b>	<b>&lt;CLEAR-TEXT&gt; –</b> , 8 16 ; <ul style="list-style-type: none"><li>• <b>encrypted</b> – :</li></ul> <b>&lt;ENCRYPTED-TEXT&gt; –</b> 8 16 ( 16 32 ) (0xYYYY...) (YYYY...).
12	IP-, SNMPv3 SNMPv3 .	<b>esr(config-snmp-user)# client-list</b> <b>&lt;NAME&gt;</b>	<b>&lt;NAME&gt; –</b> object-group, 31 .
13	IPv4/IPv6-, SNMPv3-.	<b>esr(config-snmp-user)# ip address</b> <b>&lt;ADDR&gt;</b>	<b>&lt;ADDR&gt; –</b> IP- , , AAA.BBB.CCC.DDD, [0..255].

		<b>esr(config-snmp-user)# ipv6 address &lt;ADDR&gt;</b>	<IPv6-ADDR> – IPv6-, X:X:X::X, [0..FFFF].
14	SNMPv3-	<b>esr(config-snmp-user)# enable</b>	: .
15	.	<b>esr(config-snmp-user)# privacy algorithm &lt;ALGORITHM&gt;</b>	<ALGORITHM> – : <ul style="list-style-type: none"> <li>• <b>aes 128</b> – AES-128;</li> <li>• <b>des</b> – DES.</li> </ul>
16	.	<b>esr(config-snmp-user)# privacy key ascii-text { &lt;CLEAR-TEXT&gt;   encrypted &lt;ENCRYPTED-TEXT&gt; }</b>	<CLEAR-TEXT> – , 8 16 ; <ENCRYPTED-TEXT> – 8 16 ( 16 32 ) (0xYYYY...) (YYYY...).
17	snmp view, OID user.	<b>esr(config-snmp-user)# view &lt;VIEW-NAME&gt;</b>	<VIEW-NAME> – SNMP view, OID, 31 .
18	SNMP- IP- SNMP-.	<b>esr(config)# snmp-server host { &lt;IP-ADDR&gt;   &lt;IPv6-ADDR&gt; } [vrf &lt;VRF&gt;]</b>	<IP-ADDR> – IP-, AAA.BBB.CCC.DDD, [0..255]. <IPv6-ADDR> – IPv6-, X:X:X::X, [0..FFFF]; <VRF> – VRF, SNMP-, 31 .
19	SNMP- ( ).	<b>esr(config-snmp-host)# port &lt;PORT&gt;</b>	<PORT> – UDP-, [1..65535]. : 162.
20	SNMP- .	<b>esr(config)# snmp-server enable traps &lt;TYPE&gt;</b>	<TYPE> – . . : config, entry, entry-sensor, environment, envmon, files-operations, flash, flash-operations, interfaces, links, ports, screens, snmp, syslog. .. <a href="#">ESR-Series</a> . <a href="#">CLI</a> . <a href="#">1.13</a> .
21	snmp view, OID community (SNMPv2) user (SNMPv3).	<b>esr(config)# snmp-server enable traps &lt;TYPE&gt;</b>	<VIEW-NAME> – SNMP view, 31 .

:

SNMPv3- admin. IP- esr – 192.168.52.8, IP- – 192.168.52.41.



:

:

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

:

SNMP-:

```
esr(config)# snmp-server
```

SNMPv3:

```
esr(config)# snmp-server user admin
```

:

```
esr(snmp-user)# authentication access priv
```

SNMPv3-:

```
esr(snmp-user)# authentication algorithm md5
```

SNMPv3-:

```
esr(snmp-user)# authentication key ascii-text 123456789
```

:

```
esr(snmp-user)# privacy algorithm aes128
```

:

```
esr(snmp-user)# privacy key ascii-text 123456789
```

SNMPv3-:

```
esr(snmp-user)# enable
```

- Trap-PDU-:

```
esr(config)# snmp-server host 192.168.52.41
```

## Zabbix-agent/proxy

Zabbix-agent — , , Zabbix-c. : . , , firewall — tcp, 10050. — tcp, 10051.

Zabbix- — , Zabbix-.

1	/proxy.	<b>esr(config)# zabbix-agent</b> <b>esr(config)# zabbix-proxy</b>	
2	( ). Zabbix-c.	<b>esr(config-zabbix)# hostname &lt;WORD&gt;</b> <b>esr(config-zabbix-proxy)# hostname &lt;WORD&gt;</b>	<WORD> — , 255.
3	Zabbix-c.	<b>esr(config-zabbix)# server &lt;ADDR&gt;</b> <b>esr(config-zabbix-proxy)# server &lt;ADDR&gt;</b>	<ADDR> – IP- , AAA.BBB.CCC.DDD, [0..255].
4	( ).	<b>esr(config-zabbix)# active-server &lt;ADDR&gt; &lt;PORT&gt;</b> <b>esr(config-zabbix-proxy)# active-server &lt;ADDR&gt; &lt;PORT&gt;</b>	<ADDR> – IP- , AAA.BBB.CCC.DDD, [0..255]. <PORT> – , [1..65535]. 10051.
5	, / ( ).	<b>esr(config-zabbix)# port &lt;PORT&gt;</b> <b>esr(config-zabbix-proxy)# port &lt;PORT&gt;</b>	<PORT> – , zabbix /, [1..65535]. : 10050.
6	zabbix / ( ).	<b>esr(config-zabbix)# remote-commands</b> <b>esr(config-zabbix-proxy)# remote-commands</b>	
7	, ( ).	<b>esr(config-zabbix)# source-address &lt;ADDR&gt;</b> <b>esr(config-zabbix-proxy)# source-address &lt;ADDR&gt;</b>	<ADDR> – IP- , AAA.BBB.CCC.DDD, [0..255]. : .

8	( ).	<code>esr(config-zabbix)# timeout &lt;TIME&gt;</code> <code>esr(config-zabbix-proxy)# timeout &lt;TIME&gt;</code>	<TIME> – , [1..30]. 3. , .. . , .
9	/	<code>esr(config-zabbix)# enable</code> <code>esr(config-zabbix-proxy)# enable</code>	
10	firewall ( self) TCP 10050, 10051.. <a href="#">Firewall</a>		

zabbix-agent



:

.

:

Zabbix-c, :

```
esr(config-zabbix)# server 192.168.32.101
esr(config-zabbix)# source-address 192.168.39.170
```

hostname, active-server, .

```
esr(config-zabbix)# hostname ESR-agent
esr(config-zabbix)# active-server 192.168.32.101
esr(config-zabbix)# remote-commands
```

, .

```
esr(config-zabbix)# timeout 30
esr(config-zabbix)# enable
```

zabbix-server

:

## Узлы сети

Все узлы сети / **TEST** | Активировано | **ZBX** | SNMP | JMX | IPMI | Группы элементов данных 10 | Элементы данных 94 | Триггеры 15 | Графики 36 | Правила обнаружения 2 | Веб-сценарии

Узел сети | Шаблоны | IPMI | Макросы | Инвентарные данные узла сети | Шифрование

\* Имя узла сети

ESR-agent

Видимое имя

\* Группы

SLA x

Выбрать

начните печатать для поиска

\* Должен существовать по крайней мере один интерфейс.

Интерфейсы агента

IP адрес

DNS имя

Подключаться через Порт

По умолчанию

192.168.39.170

IP

DNS

10050

Удалить

[Добавить](#)

Интерфейсы SNMP

[Добавить](#)

Интерфейсы JMX

[Добавить](#)

Интерфейсы IPMI

[Добавить](#)

Описание

Наблюдение через прокси (без прокси)

Активировано ☒

Обновить

Клонировать

Полное клонирование

Удалить

Отмена

( -> -> )



Общие
Прокси
Аутентификация
Группы пользователей
Пользователи
Способы оповещений
Скрипты
Очередь

## Скрипты

Имя

Тип

IPMIСкрипт

Выполнять на

Zabbix агентZabbix сервер (прокси)Zabbix сервер

Команды

zabbix\_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo ping -c 3 192.168.32.101]"

Описание

Группа пользователей

Все

Группа узлов сети

Все

Требуемые права доступа к узлам сети

ЧтениеЗапись

Включить подтверждение

☐

Текст подтверждения

Тест подтверждения

Обновить

Клонировать

Удалить

Отмена

ESR :

- **Ping:**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo ping -c 3 192.168.32.101]"
```

(ESR), , ping ( 192.168.32.101), .

"-c" -.. ping .

- **Ping VRF:**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[sudo netns -exec -n backup sudo ping 192.168.32.101 -c 5 -W 2 ]"
```

VRF backup.

- **Fping**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo fping 192.168.32.101]"
```

(ESR), , fping ( 192.168.32.101), .

- **Fping VRF**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[sudo netns-exec -n backup sudo fping 192.168.32.101 ]"
```

- **Traceroute**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo traceroute 192.168.32.101]"
```

(ESR), , traceroute ( 192.168.32.101), .

- **Traceroute VRF**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo netns-exec -n backup sudo traceroute 192.168.32.179]"
```

- **Iperf**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo iperf -c 192.168.32.101 -u -b 100K -i 1 -t 600]"
```

(ESR), , iperf ( 192.168.32.101), .

- **Iperf VRF**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[ sudo netns-exec -n backup sudo iperf -c 192.168.32.101 -u -b 100K -i 1 -t 600]"
```

- **Nslookup**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[sudo nslookup ya.ru ]"
```

(ESR), , nslookup , .

- **Nslookup VRF**

```
zabbix_get -s {HOST.CONN} -p 10050 -k "system.run[sudo netns-exec sudo nslookup ya.ru ]"
```

Iperf:

#### iperf\_agent

```
zabbix_get -s 192.168.39.170 -p 10050 -k "system.run[ sudo iperf -c 192.168.32.101]"
```

```
-----  
Client connecting to 192.168.32.101, TCP port 5001  
TCP window size: 49.5 KByte (default)  
-----
```

```
[ 3] local 192.168.39.170 port 52815 connected with 192.168.32.101 port 5001  
[ ID] Interval      Transfer    Bandwidth  
[ 3]  0.0-10.0 sec  1.01 GBytes  864 Mbits/sec
```

Отмена

, , : snmpget, cat, pwd, wget .

snmpget:

#### snmpget\_Des

```
zabbix_get -s 192.168.39.230 -p 10050 -k "system.run[snmpget -v 2c -c public localhost .1.3.6.1.2.1.1.1.0 ]"
```

```
.1.3.6.1.2.1.1.1.0 = STRING: "Eltex ESR-1200 Service Router 1.14.x build 7 (date 15/10/2020 time 23:13:19)"
```

Отмена

Syslog

Syslog (. *System Log*— ) — , , IP.

1	syslog-, snmp- snmp-trap.	esr(config)# syslog snmp <SEVERITY>	<SEVERITY> – , ( ); <ul style="list-style-type: none"><li>• emerg – , ;</li><li>• alert – , ;</li><li>• crit – , ;</li><li>• error – ;</li><li>• warning – , ;</li><li>• notice – ;</li><li>• info – ;</li><li>• debug – , ;</li><li>• none – syslog-.</li></ul>
2	syslog-, (Telnet, SSH) ( )	esr(config)# syslog monitor <SEVERITY>	
3	syslog- ( )	esr(config)# syslog cli-commands	
4	syslog	esr(config)# syslog file <NAME> <SEVERITY>	<NAME> – , , 31; <SEVERITY> syslog snmp.
5	( )	esr(config)# syslog file-size <SIZE>	<SIZE> – , [10..10000000]
6	, ( )	esr(config)# syslog max-files <NUM>	<NUM> – , [1.. 1000]
7	syslog syslog-	esr(config)#syslog host <HOSTNAME> <ADDR> <SEVERITY> <TRANSPORT> <PORT>	<HOSTNAME> – syslog-, 31 . . «all» no syslog host syslog-; <ADDR> – IP-, AAA.BBB.CCC.DDD, [0..255]; <SEVERITY> – , , Syslog; <TRANSPORT> – , , : <ul style="list-style-type: none"><li>• TCP – TCP;</li><li>• UDP – UDP;</li></ul> <PORT> – TCP/UDP-, , [1..65535], 514
8	( )	esr(config)#syslog reload debugging	
9	( )	esr(config)#syslog sequence-numbers	
10	( ).	esr(config)#syslog timestamp msec	
11	( ).	esr(config)#logging login on-failure	
12	( ).	esr(config)#logging syslog configuration	
13	( ).	esr(config)#logging userinfo	

:  
:  

- ;
- ;
- / ;
- .

IP- ESR – 192.168.52.8, ip- Syslog – 192.168.52.41. – UDP 514.



:  
:  

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

:  
  
, - info:

```
esr(config)# syslog file ESR info
```

IP Syslog-:

```
esr(config)# syslog host SERVER 192.168.17.30 info udp 514
```

:

```
esr(config)# logging login on-failure
```

syslog:

```
esr(config)# logging syslog configuration
```

/:

```
esr(config)# logging service start-stop
```

:

```
esr(config)# logging userinfo
```

:

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

:

```
esr# show syslog configuration
```

:

```
esr# show syslog ESR
```

.

1		esr# verify filesystem <detailed>	detailed -- .

:

:

:

:

```
esr# verify filesystem
Filesystem Successfully Verified
```

ESR / .

1		esr(config)# archive	
2	()	esr(config-ahchive)# type <TYPE>	<TYPE> - . :  • local; • remote; • both.  : remote
3	()	esr(config-ahchive)# auto	
4	()	esr(config-ahchive)# by-commit	
5	( remote both)	esr(config-ahchive)# path <PATH>	<PATH> - , ,
6	( , auto)	esr(config-ahchive)# time-period <TIME>	<TIME> - , [1..35791394].  : 720
7	( , local both)	esr(config-ahchive)# count-backup <NUM>	<NUM> - . [1..100].  : 1

:

1 . tftp- 172.16.252.77 esr-example. - 30.

:

, IP-, tftp- .

:

:

```
esr# configure
esr(config)# archive
```

:

```
esr(config)# type both
```

:

```
esr(config-archive)# path tftp://172.16.252.77:/esr-example/esr-example.cfg
esr(config-archive)# count-backup 30
```

:

```
esr(config-archive)# time-period 1440
```

:

```
esr(config-archive)# auto  
esr(config-archive)# by-commit
```

```
1      tftp-      "esr-exampleYYYYMMDD_HHMMSS.cfg".,      flash:backup/      "config_YYYYMMDD_HHMMSS". flash:backup/ 30 ,      .
```