AMRES
Akademska mreža Srbije

Uputstvo za FreeRADIUS i LDAP konfiguraciju

Istorija verzija dokumenta

Verzija	Datum	Inicijali autora	Opis promene
1.0	27.04.2016.	ME	Prva verzija dokumenta

Sadržaj

1 Konfiguracija proxy.conf fajla

Potrebno je preći u raddb direktorijum:

```
cd /usr/local/etc/raddb
```

Nakon toga, u okviru proxy.conf fajla dodati domen za vašu instituciju na kraj ovog fajla:

```
realm inst.ac.rs {
        pool = inst.ac.rs
}
home_server_pool inst.ac.rs {
        home_server = localhost
}
```

pri čemu je inst.ac.rs potrebno zameniti domenom vaše institucije (npr. amres.ac.rs, fon.bg.ac.rs i sl.). Kada napraviti neophodne izmene, snimite ih i izađite iz proxy.conf fajla. Proverite da li su sve izmene u redu pomoću komande:

```
cat proxy.conf | grep -v "#" | less
```

2 Konfiguracija clients.conf fajla

U okviru raddb direktorijuma nalazi se i clients.conf fajl. Potrebno je na kraj fajla dodati sve unose za RADIUS klijente koji su dati u nastavku.

```
## eduroam Federation Top Level Radius serveri:
##eduroam ftlr1
client ftlr1.ac.rs {
 ipaddr = 147.91.4.204
 secret = pass # - lozinka se dobija od AMRES-a
 shortname = ftlr1
 nastype = other
 virtual_server = eduroam
}
##eduroam ftlr2
client ftlr2.ac.rs {
    ipaddr = 147.91.1.101
    secret = pass # - lozinka se dobija od AMRES-a
    shortname = ftlr2
    nastype = other
    virtual_server = eduroam
}
##Monitoring eduroam servisa
```

```
client netiis.monitor {
    ipaddr = 147.91.3.12
    secret = pass # - lozinka se dobija od AMRES-a
    shortname = netiis
    nastype = other
    virtual_server = eduroam
}
# Test FTRL
client FTLR-test {
    ipaddr = 192.168.100.30
    secret = test123
    shortname = Ftlr
    nastype = other
    virtual_server = eduroam
}
```

Proveriti da li su sve izmene ispravno napravljene korišćenjem iste komande kao u Appendix A sekciji.

```
cat clients.conf | grep -v "#" | less
```

3 Kreiranje eduroam i eduroam-inner-tunnel virtuelnih servera

Preći u poddirektorijum sites-available:

```
cd /usr/local/etc/raddb/sites-available
```

Sada je potrebno kopirati default virtuelni server u eduroam virtuelni server:

```
cp default eduroam
```

Nakon toga, napraviti izmene tako da fajl izgleda kao u nastavku:

```
server eduroam {
listen {
        type = auth
        ipaddr = *
        port = 0
        limit {
              max_connections = 16
              lifetime = 0
              idle_timeout = 30
        }
}
```

```
listen {
        ipaddr = *
        port = 0
        type = acct
        limit {
        }
}

listen {
        type = auth
        port = 0
        limit {
              max_connections = 16
              lifetime = 0
              idle_timeout = 30
        }
}

listen {
        ipv6addr = ::
        port = 0
        type = acct
        limit {
        }
}

authorize {
        filter_username
        preprocess
        auth_log
        eap {
                ok = return
        }
        expiration
        logintime
}
authenticate {
```

```
        Auth-Type PAP {
                pap
        }

        Auth-Type CHAP {
                chap
        }

        Auth-Type MS-CHAP {
                mschap
        }

        digest
        eap
}

preacct {
        preprocess
        acct_unique
        suffix
        files
}

accounting {
        detail
        exec
        attr_filter.accounting_response
}

session {
}

post-auth {
        update {
                &reply: += &session-state:
        }
        reply_log
        exec
```

```
        remove_reply_message_if_eap
        Post-Auth-Type REJECT {
                attr_filter.access_reject
                eap
                remove_reply_message_if_eap
        }
}

pre-proxy {
}

post-proxy {
        eap
}
}
```

Na sličan način se kreira i eduroam-inner-tunnel virtuelni server:

```
cp inner-tunnel eduroam-inner-tunnel
```

Sledeći korak je izmena konfiguracije ovog virtuelnog servera tako da se dobije konfiguracija kao u nastavku:

```
server eduroam-inner-tunnel {
authorize {
        filter_username
        chap
        mschap
        suffix
        update control {
                &Proxy-To-Realm := LOCAL
        }
        files
        -ldap
        pap
}

authenticate {
        Auth-Type PAP {
                pap
        }
```

```
        Auth-Type CHAP {
                chap
        }

        Auth-Type MS-CHAP {
                mschap
        }

        eap
}

session {
}

post-auth {
        reply_log
        Post-Auth-Type REJECT {
           attr_filter.access_reject
            update outer.session-state {
              &Module-Failure-Message := &request:Module-Failure-Message
            }
          }
}

pre-proxy {
}

post-proxy {
        eap
}
}
```

NAPOMENA: Posebno obratiti pažnju na osenčene linije. Proveriti kako izgleda fajl bez komentara nakon što se unesu izmene.

Ova dva virtuelna servera se i dalje ne mogu koristiti. Da bi ih server učitao prilikom pokretanja, potrebno je napraviti soft linkove ka ovim virtuelnim serverima. Prvo se prelazi u sites-enabled direktorijum:

```
cd ../sites-enabled
```

Kako bi se kreirali soft linkovi, unose se sledeće komande:

```
ln -s ../sites-available/eduroam
ln -s ../sites-available/eduroam-inner-tunnel
```

4 Konfigurisanje FreeRADIUS eap modula

Najveći broj modula se nalazi u mods-available poddirektorijumu:

```
cd ../mods-available
```

Modul koji je odgovoran za uspostavljanje sigurnog tunela je eap. Izmene koje je potrebno napraviti su date u nastavku. Nakon napravljenih izmena, snimiti fajl i proveriti da li je sve konfigurisano ispravno.

```
eap {
        default_eap_type = ttls
        timer_expire     = 60
        ignore_unknown_eap_types = no
        cisco_accounting_username_bug = no
        max_sessions = ${max_requests}
        md5 {
        }
        leap {
        }
        gtc {
                auth_type = PAP
        }

        tls-config tls-common {
                private_key_password = whatever
                private_key_file = ${certdir}/server.pem
                certificate_file = ${certdir}/server.pem
                ca_file = ${cadir}/ca.pem
                dh_file = ${certdir}/dh
                ca_path = ${cadir}
                cipher_list = "DEFAULT"
        }
        tls {
                tls = tls-common
        }
        ttls {
                tls = tls-common
                default_eap_type = md5
```

```
                copy_request_to_tunnel = no
                use_tunneled_reply = no
                virtual_server = "eduroam-inner-tunnel"
        }
        peap {
                tls = tls-common
                default_eap_type = mschapv2
                copy_request_to_tunnel = no
                use_tunneled_reply = no
                virtual_server = "inner-tunnel"
        }
        mschapv2 {
        }
}
```

Kako bi eap modul bio učitan prilikom pokretanja RADIUS servera, i za njega je potrebno napraviti soft link, ali u mods-enabled direktorijumu:

```
ln -s ../mods-available/eap
```

5 Konfigurisanje OpenLDAP servera

Konfiguracioni fajl koji se koristi za zadavanje početnih parametara LDAP serveru se nalazi u openldap direktorjimu. Puna putanja je:

```
cd /usr/local/etc/openldap
```

Ime fajla koji se menja je slapd.conf i treba da izgleda kao u nastavku:

```
    include         /usr/local/etc/openldap/schema/core.schema
    include         /usr/local/etc/openldap/schema/cosine.schema
    include         /usr/local/etc/openldap/schema/inetorgperson.schema
    include         /usr/local/etc/openldap/schema/nis.schema
    pidfile         /usr/local/var/run/slapd.pid
    argsfile        /usr/local/var/run/slapd.args
    database        mdb
    maxsize         1073741824
    suffix          "dc=inst,dc=ac,dc=rs"
    rootdn          "cn=Manager,dc=inst,dc=ac,dc=rs"
    rootpw          secret
    directory       /usr/local/var/openldap-data
    index   objectClass     eq
```

Osenčene su linije koje je potrebno izmeniti. Parametar suffix u stvari predstavlja domen vaše institucije. LDAP je specifičan po tome što dozvoljava kreiranje svog root korisnika, čiji parametri su rootdn i rootpw. Prilikom izbora lozinke, vodite računa jer nisu podržani svi specijlani znakovi. Inicijalno LDAP stablo se kreira uz pomoć **inst.ac.rs.ldif** fajla koji se nalazi u istom direktorijumu kao i konfiguracioni fajl **slapd.conf**. U inst.ac.rs.ldif fajlu je potrebno napraviti izmene na osenčenim mestima u skladu sa prethodno definisanim parametrima iz slapd.conf fajla.

```
dn: ou=People,dc=inst,dc=ac,dc=rs
objectClass: top
objectClass: organizationalUnit
ou: People

dn: uid=test,ou=People,dc=inst,dc=ac,dc=rs
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
cn: Test
sn: Test
uid: test
userPassword:{SHA}cojt0Pw//L6ToM8G41aOKFIWh7w=
```

Kada su svi parametri u slapd.conf inst.ac.rs.ldif fajlovima ispravno podešeni, može se preći na ubacivanje inicijalnog inst.ac.rs.ldif fajla u LDAP, komandom:

```
ldapadd -f inst.ac.rs.ldif -D rootdn -w rootpw
```

Osenčeni su parametri koje je potrebno promeniti. Inicijalni fajl koji je prethodno kreiran i koji se ubacuje u LDAP mora imati ***.ldif*** ekstenziju. Parametar **rootdn** odgovara superadmin nalogu iz slapd.conf fajla, a parametar **password** je lozinka za ovaj nalog. Pre nego što početno stablo može da se doda u LDAP direktorjium, LDAP server mora biti pokrenut. Komanda koja ovo omogćava je:

```
/usr/local/libexec/slapd
```

Na ovaj način će biti kreirano LDAP stablo sa korenom čije ime će odgovarati domenu institucije, sa jednom granom (People) i jednim korisničkim nalogom (test). U koren stabla se naknadno mogu dodavati i nove grane, bilo iz komandne linije, bilo uz korišćenje Apache Directory Studio softvera.

6 Konfiguracija FreeRADIUS ldap modula

LDAP modul se, kao i EAP modul, nalazi u mods-available direktorijumu. Prve tri osenčene linije koje omogućavaju komunikaciju sa LDAP direktorijumom je potrebno popuniti tako da odgovaraju parametrima iz slapd.conf fajla. Od ostalih parametara jedino se mora dodati kontrolni parametar za korisničko ime u update sekciji:

```
ldap {
        server = 'localhost'
        identity ='cn=Manager,dc=inst,dc=ac,dc=rs'
```

```
        password = mypass
        base_dn = 'dc=inst,dc=ac,dc=rs'
        sasl {
        }
        update {
                control:Password-With-Header    += 'userPassword'
                control:Stripped-User-Name      := 'uid'
                control: += 'radiusControlAttribute'
                request: += 'radiusRequestAttribute'
                reply: += 'radiusReplyAttribute'
        }
        user {
                base_dn = "${..base_dn}"
                filter = "(uid=%{%{Stripped-User-Name}:-%{User-Name}})"
                sasl {
                }
        }
        group {
                base_dn = "${..base_dn}"
                filter = '(objectClass=posixGroup)'
                membership_attribute = 'memberOf'
        }
        profile {
        }
        client {
                base_dn = "${..base_dn}"
                filter = '(objectClass=radiusClient)'
                template {
                }
                attribute {
                        ipaddr = 'radiusClientIdentifier'
                        secret = 'radiusClientSecret'
                }
        }
        accounting {
                reference = "%{tolower:type.%{Acct-Status-Type}}"
                type {
                        start {
```

```
                        update {
                                description := "Online at %S"
                        }
                }
                interim-update {
                        update {
                                description := "Last seen at %S"
                        }
                }
                stop {
                        update {
                                description := "Offline at %S"
                        }
                }
        }
}

post-auth {
        update {
                description := "Authenticated at %S"
        }
}

options {
        chase_referrals = yes
        rebind = yes
        res_timeout = 10
        srv_timelimit = 3
        net_timeout = 1
        idle = 60
        probes = 3
        interval = 3
        ldap_debug = 0x0028
}
tls {
}
pool {
        start = ${thread[pool].start_servers}
```

```
                min = ${thread[pool].min_spare_servers}
                max = ${thread[pool].max_servers}
                spare = ${thread[pool].max_spare_servers}
                uses = 0
                retry_delay = 30
                lifetime = 0
                idle_timeout = 60
        }
}
```

Kao i prilikom prethodnih izmena, ponovo proverite da li fajl na serveru izgleda kao u ovom dokumentu.

7 Testiranje autentifikacije pomoću eapol_test programa

Kada su svi prethodni fajlovi konfigurisani, završena je konfiguracija RADIUS servera i LDAP direktorijuma. Bilo koja naredna promena u konfiguraciji podrazumeva da se RADIUS proces mora zaustaviti i ponovo pokrenuti. Najbolji nači da se utvrdi da li se konfiguracija učitala bez problema i da li možda postoje neka upozorenja je da se RADIUS server prvo pokrene u ***debug*** modu. U komandnoj liniji je potrebno uneti:

```
radiusd -X
```

Ukoliko se konfiguracija učitala bez problema, poslednja linija koja se ispisuje je:

```
Ready to process requests
```

Nakon toga, preporučljivo je da se testira autentifikacija. Ovo je moguće uraditi na dva načina:

- primenom **eapol_test** alata i
- povezivanjem preko AP-a koji se koristi u okviru eduroam servisa.

Pre samog testiranja, neophodno je podesiti lokalni firewall na Linux operativnom sistemu u **iptables** fajlu kako bi virtuelna mašina dozvolila komunikaciju po željenim portovima. Ovaj fajl se nalazi u /etc/sysconfig direktorijum i mora dozvoliti komunikaciju po RADIUS portovima 1812, 1813 i 1814. Potrebno je u već postojeća pravila dodati sledeće linije:

```
# FTLR1
-A INPUT -m state -s 147.91.4.204/32 --state NEW -m udp -p udp --dport 1812 -j ACCEPT
-A INPUT -m state -s 147.91.4.204/32 --state NEW -m udp -p udp --dport 1813 -j ACCEPT
-A INPUT -m state -s 147.91.4.204/32 --state NEW -m udp -p udp --dport 1814 -j ACCEPT
# FTLR2
-A INPUT -m state -s 147.91.1.101/32 --state NEW -m udp -p udp --dport 1812 -j ACCEPT
-A INPUT -m state -s 147.91.1.101/32 --state NEW -m udp -p udp --dport 1813 -j ACCEPT
-A INPUT -m state -s 147.91.1.101/32 --state NEW -m udp -p udp --dport 1814 -j ACCEPT
# NetIIS
-A INPUT -m state -s 147.91.3.12/32 --state NEW -m udp -p udp --dport 1812 -j ACCEPT
-A INPUT -m state -s 147.91.3.12/32 --state NEW -m udp -p udp --dport 1813 -j ACCEPT
-A INPUT -m state -s 147.91.3.12/32 --state NEW -m udp -p udp --dport 1814 -j ACCEPT
# Test FTLR
-A INPUT -m state -s 192.168.100.30/32 --state NEW -m udp -p udp --dport 1812 -j ACCEPT
```

```
-A INPUT -m state -s 192.168.1.30/32 --state NEW -m udp -p udp --dport 1813 -j ACCEPT
-A INPUT -m state -s 192.168.1.30/32 --state NEW -m udp -p udp --dport 1814 -j ACCEPT
```

Za oba testa je potrebno prethodno napraviti test korisnika u LDAP direktorijmu koji je kreiran. Alat koji omogućava testiranje iz komandne linije i bez povezivanja na AP-ove je **eapol_test**. Ovaj alat simulira vezu AP/server, tj. predstavlja se serveru kao klijent koji je u stvari AP. Komanda koja omogućava testiranje je:

```
eapol_test -c ttls-pap.conf -s testing123
```

pri čemu ***-c*** definiše odakle će se čitati parametri korisničkog naloga, a ***-s*** predstavlja **shared_secret** lozinku za komunikaciju sa RADIUS klijentom. U ovom slučaju, RADIUS klijent kome se zahtev prosleđuje je **localhost** klijent definisan u **clients.conf** fajlu. Ovaj alat je već instaliran na serveru, pa je potrebno samo izmeniti parametre u **ttls-pap.conf** fajlu. Puna putanja do fajla je

```
cd /opt/
```

Izgled ovog fajla je dat u nastavku, a osenčene linije predstavljaju onaj deo koji je potrebno izmeniti tako da odgovaraju test nalogu iz LDAP-a. Vrednosti parametra **anonymous_identity** treba samo dodati domen vaše institucije.

```
network={
        ssid="example"
        key_mgmt=WPA-EAP
        eap=TTLS
        identity="bob"
        anonymous_identity="anonymous"
        password="hello"
        phase2="auth=PAP"
}
```

Prva faza testiranja obuhvata testiranje sa tekstualnim fajlom. Na kraj users fajla u raddb direktorijumu je potrebno dodati sledeću liniju:

```
test Password-With-Header := "{SHA}cojt0Pw//L6ToM8G41aOKFIWh7w="
```

Tek nakon što promenite parametre iz ttls-pap.conf fajla i dodate test korisnika u users fajl, možete preći na testiranje. Pre pokretanja eapol_test komande, otvorite drugi Putty prozor u kom ćete pokrenuti FreeRADIUS u debug modu.

Druga faza testiranja se bazira na proveri kredencijala test korisnika, pri čemu su sada oni smešteni u LDAP direktorijum. Stoga se u **eduroam-inner-tunnel** virtuelnom serveru, u okviru **authorize** sekcije, zakomentarisati opcija **files** dodavanjem znaka „#" ispred, a otkomentarisati opcija **ldap**. Sada je potrebno ponovo pokrenuti server u debug modu kako bi se proverilo da li postoji komunikacija između FreeRADIUS servera i OpenLDAP servera. Ukoliko je sve u redu, ponoviti postupak testiranja iz prethodnog koraka.

8 Dodavanje korisnika u LDAP direktorijum uz pomoć Apache Directory Studio programa

Dodavanje novog korisnika u LDAP može biti nezgrapno ukoliko se za to koristi komandna linija. Zbog toga je najbolje koristiti npr. Apache Directory Studio softver.

Novi korisnik se dodaje u nekoliko koraka:

- Desnim klikom na **uid=test** se otvara novi meni iz koga je potrebno odabrati opciju „**Copy Entry/DN**"
- Desni klik na People granu daje, između ostalih, i opciju „**Paste Entry**"
- Odabrati opciju „**Rename entry and continue**", a u delu **RDN: uid = test**, potrebno je dodeliti novu vrednost, npr. **uid = test2** i kliknuti na **dugme „OK"**

Nakon koraka 3 se u stablu može videti novi korisnik, odnosno uid koji je prethodno kreiran.

9 Tumačenje log poruka iz radius.log fajla

Konfiguracioni fajl koji utiče na ponašanje RADIUS servera u celini je radiusd.conf. Za neke parametre se koriste uobičajene vrednosti, koje ne treba menjati osim u slučaju da ste sigurni šta menjate i koje su posledice. Takođe, ovaj fajl omogućava definisanje parametara koji će kao posledicu imati upis nekih informacija od značaja u radius.log fajl. Lokacija ovog fajla je /usr/local/var/log/radius. Posebno je bitna log sekcija, koju je potrebno promeniti tako da izgleda kao u nastavku.

```
log {
        destination = files
        file = ${logdir}/radius.log
        syslog_facility = daemon
        stripped_names = no
        auth = yes
        auth_badpass = no
        auth_goodpass = no
}
```

Opcija „auth=yes" omogućava da se pored anonimnog identiteta u log fajlovima beleže i prava korisnička imena, iz unutrašnjeg tunela. Ovaj podatak je bitan kako bi se mogao mapirati korisnik u slučaju izazivanja nekog sigurnosnog incidenta ili npr. prilikom kršenja autorskih prava („torrent"). Posebno je potrebno obratiti pažnju na opcije **auth_badpass** i **auth_goodpass**. Ove opcije omogućavaju logovanje lozinki korisnika u slučaju kada se uspešno/neuspešno autentifikuju. Kako bi se zaštitila privatnost korisnika, ovu opciju **nikada** nemojte postavljati na vrednost „yes".

U zavisnosti od toga da li je korisnik pogrešio korisničko ime ili lozinku, drugačiji će biti ispis u logovima. Primer log linija za neuspešnu autentifikaciju, u slučaju kada korisničko ime ne postoji (npr. **mark@amres.ac.rs** umesto **marko@amres.ac.rs**):

```
Login incorrect: [anonymous@amres.ac.rs] (from client ac.rs port 2 cli 10-0b-a9-39-28-ec)
Login incorrect ([ldap] User not found): [mark@amres.ac.rs] (from client ac.rs port 0 via TLS tunnel)
```

Prva linija je sa korisničkim imenom van tunela i može se videti MAC adresa korisničkog uređaja, dok druga linija prikazuje korisničko ime unutar tunela.

Ukoliko je korisničko ime ispravno, ali je korisnik pogrešio lozinku, u log fajlu se ispisuje sledeća linije:

```
Login incorrect: [anonymous@amres.ac.rs] (from client ac.rs port 2 cli 10-0b-a9-39-28-ec)
Login incorrect (rlm_pap: SHA1 password check failed): [marko@amres.ac.rs] (from client ac.rs port 0 via TLS tunnel)
```