Configuring PEAP / LDAP based authentication using FreeRADIUS on Debian Sarge and Cisco AP1200, with WPA2 AES encryption

Ivan Klimek
Computer Networks Laboratory
Technical University Kosice, Slovakia
http://www.cnl.tuke.sk

1. Introduction

This document describes the configuration steps needed to set up and use 802.1X: Port-Based Network Access Control using PEAP (PEAP/MS-CHAPv2) as authentication method and FreeRADIUS as back-end authentication server running on Debian Sarge. Cisco AP1200 series as the authenticator. And Windows XP default build-in supplicant.

2. FreeRadius

2.1 Before the installation

apt-get install libssl-dev
apt-get build-dep freeradius

2.2 Working with the source

- getting the source code:
apt-get source freeradius

- unpacking, compile, make, make install (the filename can be different)

tar zxfv freeradius-1.0.4.tar.gz
cd freeradius-1.0.4

./configure --disable-shared
make
make install

2.3 Configuring FreeRADIUS

- the binaries are installed in /usr/local/bin and /usr/local/sbin. The configuration files are found under /usr/local/etc/raddb.

cd /usr/local/etc/raddb

- Open the main configuration file radiusd.conf. Inside the encrypted PEAP tunnel, an MS-CHAPv2 authentication mechanism is used.

- it should look like this:

```
mschap {
        #
        #  As of 0.9, the mschap module does NOT support
        #  reading from /etc/smbpasswd.
        #
        #  If you are using /etc/smbpasswd, see the 'passwd'
        #  module for an example of how to use /etc/smbpasswd

        # authtype value, if present, will be used
        # to overwrite (or add) Auth-Type during
        # authorization. Normally should be MS-CHAP
        authtype = MS-CHAP

        # if use_mppe is not set to no mschap will
        # add MS-CHAP-MPPE-Keys for MS-CHAPv1 and
        # MS-MPPE-Recv-Key/MS-MPPE-Send-Key for MS-CHAPv2
        #
        use_mppe = yes

        # if mppe is enabled require_encryption makes
        # encryption moderate
        #
        require_encryption = yes

        # require_strong always requires 128 bit key
        # encryption
        #
        require_strong = yes

        # Windows sends us a username in the form of
        # DOMAIN\user, but sends the challenge response
        # based on only the user portion.  This hack
        # corrects for that incorrect behavior.
        #
        #with_ntdomain_hack = no

        # The module can perform authentication itself, OR
        # use a Windows Domain Controller.  This configuration
        # directive tells the module to call the ntlm_auth
        # program, which will do the authentication, and return
        # the NT-Key.  Note that you MUST have "winbindd" and
```

```
                # "nmbd" running on the local machine for ntlm_auth
                # to work.  See the ntlm_auth program documentation
                # for details.
                #
                # Be VERY careful when editing the following line!
                #
                #ntlm_auth = "/path/to/ntlm_auth --request-nt-key --username=%
{Stripped-User-Name:-%{User-Name:-None}} --challenge=%{mschap:Challenge:-
00} --nt-response=%{mschap:NT-Response:-00}"
        }
```

- configuring LDAP support:

```
# Lightweight Directory Access Protocol (LDAP)
        #
        #  This module definition allows you to use LDAP for
        #  authorization and authentication (Auth-Type := LDAP)
        #
        #  See doc/rlm_ldap for description of configuration options
        #  and sample authorize{} and authenticate{} blocks
        ldap {
                server = "10.0.0.4"
            identity = "cn=wifiadmin-ro,cn=ServiceAdmins,ou=LdapAdmins,dc=sk"
            password = "password"
            basedn = "dc=wifi.cnl.tuke.sk,ou=People,dc=sk"
            filter = "(eapLogin=%{Stripped-User-Name:-%{User-Name}})"
            start_tls = no
            tls_mode = no

            dictionary_mapping = ${raddbdir}/ldap.attrmap

            ldap_connections_number = 5
            password_attribute = eapUserPassword
            reply_attribute = eapUserPassword
            timeout = 4
            timelimit = 3
            net_timeout = 1

        }
```

- Also make sure the "authorize" and "authenticate" contains:

```
authorize {
        preprocess
        mschap
        suffix
        eap
        files
        ldap
```

```
}


authenticate {

        #
        #  MSCHAP authentication.
        Auth-Type MS-CHAP {
             mschap
         }

          #
        #  Allow EAP authentication.
        eap
    }
```

- open the file clients.conf which speciefies the AP the authenticator server will be serving, a sample:

```
client 10.0.0.1 {
secret = secret
shortname = test
}
```

- this specifies the IP address of the AP. Secret stands for shared secret, thats the password needed for communication between the authenticator (AP) and the authenticator server (FreeRadius). Shortname is meaningless, but cannot be omitted.

- proceed to the file eap.conf, in the eap section, set default eap type to peap:

```
default_eap_type = peap
```

- since PEAP is using TLS, the TLS section must contain:

```
tls {
                        private_key_password = whatever
                        private_key_file = ${raddbdir}/certs/cert-srv.pem

                        #  If Private key & Certificate are located in
                        #  the same file, then private_key_file &
                        #  certificate_file must contain the same file
                        #  name.
                        certificate_file = ${raddbdir}/certs/cert-srv.pem
```

```
                # Trusted Root CA list
                CA_file = ${raddbdir}/certs/demoCA/cacert.pem

                dh_file = ${raddbdir}/certs/dh
                random_file = ${raddbdir}/certs/random
                }
```

- it isnt necessary to create new certificates, nobody will ever see them. If you set the *private_key_password = whatever* and uncomment all lines as shown in the sample output, it will work, since FreeRadius comes with some built-in certificates. But if you want to create our own certificates, dont use the CA.all scripts, try tinyCA. Dont forget to specify the password in the certificate the same as here in eap.conf.

- find the "peap" section, and make sure it contain the following:

```
peap {
                # The tunneled EAP session needs a default
                # EAP type which is separate from the one for
                # the non-tunneled EAP module. Inside of the
                # PEAP tunnel, we recommend using MS-CHAPv2,
                # as that is the default type supported by
                # Windows clients.
                default_eap_type = mschapv2
        }
```

3. Cisco AP1200

- sample config:

```
version 12.3
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname CNL-test-AP
!
enable secret 5 $1$xRPb$3EqSNTpcS0SfFQMZ/15Gs1
!
ip subnet-zero
!
aaa new-model
!
aaa group server radius rad_eap
 server 10.0.0.3 auth-port 1812 acct-port 1813
!
```

```
aaa group server radius rad_mac
!
aaa group server radius rad_acct
!
aaa group server radius rad_admin
 cache expiry 1
 cache authorization profile admin_cache
 cache authentication profile admin_cache
!
aaa group server radius rad_pmip
!
aaa group server radius dummy
!
aaa authentication login eap_methods group rad_eap
aaa authentication login eap_methods group rad_eap
aaa authorization exec default local
aaa accounting network acct_methods start-stop group rad_acct
aaa cache profile admin_cache
 all
!
aaa session-id common
!
dot11 ssid CNL-KPI-240
   vlan 240
   authentication open eap eap_methods
   authentication network-eap eap_methods
   authentication key-management wpa
!
dot11 ssid CNL-KPI-PDA
   vlan 666
   authentication open eap eap_methods
   authentication network-eap eap_methods
   authentication key-management wpa
!
dot11 ssid CNL-KPI-PEAP-WPA2
   vlan 481
   authentication open eap eap_methods
   authentication network-eap eap_methods
   authentication key-management wpa
   guest-mode
!
!
!
username Cisco password 7 14341B180F0B
!
bridge irb
!
!
interface Dot11Radio0
 no ip address
 no ip route-cache
 !
 encryption vlan 481 mode ciphers aes-ccm
 !
 encryption vlan 240 mode ciphers aes-ccm
 !
 encryption vlan 666 mode ciphers tkip
 !
 ssid CNL-KPI-240
 !
 ssid CNL-KPI-PDA
```

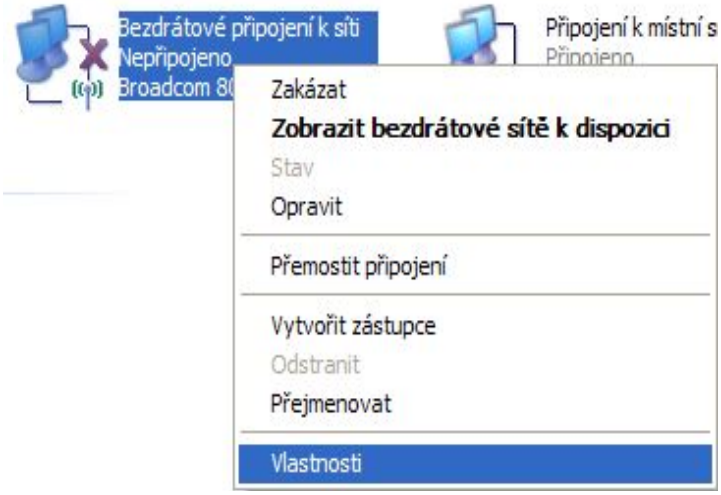
```
 !
 ssid CNL-KPI-PEAP-WPA2
 !
 speed basic-1.0 2.0 5.5 6.0 9.0 11.0 12.0 18.0 24.0 36.0 48.0 54.0
 station-role root
 bridge-group 1
 bridge-group 1 block-unknown-source
 no bridge-group 1 source-learning
 no bridge-group 1 unicast-flooding
 bridge-group 1 spanning-disabled
!
interface Dot11Radio0.240
 encapsulation dot1Q 240
 no ip route-cache
 bridge-group 240
 bridge-group 240 subscriber-loop-control
 bridge-group 240 block-unknown-source
 no bridge-group 240 source-learning
 no bridge-group 240 unicast-flooding
 bridge-group 240 spanning-disabled
!
interface Dot11Radio0.481
 encapsulation dot1Q 481
 no ip route-cache
 bridge-group 255
 bridge-group 255 subscriber-loop-control
 bridge-group 255 block-unknown-source
 no bridge-group 255 source-learning
 no bridge-group 255 unicast-flooding
 bridge-group 255 spanning-disabled
!
interface Dot11Radio0.666
 encapsulation dot1Q 666
 no ip route-cache
 bridge-group 254
 bridge-group 254 subscriber-loop-control
 bridge-group 254 block-unknown-source
 no bridge-group 254 source-learning
 no bridge-group 254 unicast-flooding
 bridge-group 254 spanning-disabled
!
interface FastEthernet0
 no ip address
 no ip route-cache
 duplex auto
 speed auto
 bridge-group 1
 no bridge-group 1 source-learning
 bridge-group 1 spanning-disabled
!
interface FastEthernet0.240
 encapsulation dot1Q 240
 no ip route-cache
 bridge-group 240
 no bridge-group 240 source-learning
 bridge-group 240 spanning-disabled
!
interface FastEthernet0.481
 encapsulation dot1Q 481
 no ip route-cache
 bridge-group 255
```

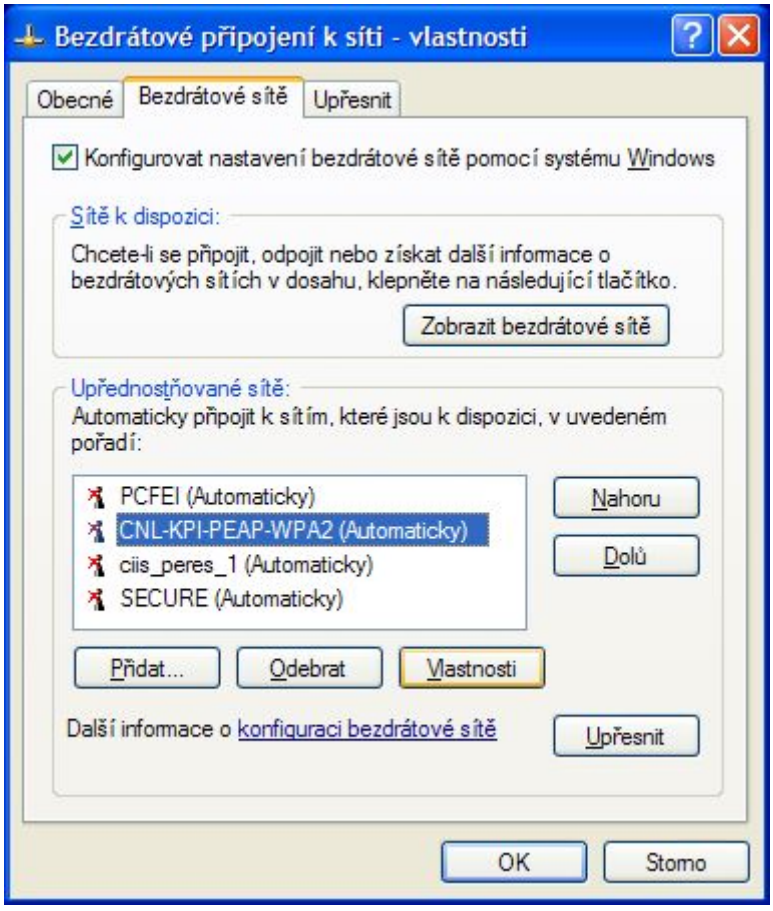
```
 no bridge-group 255 source-learning
 bridge-group 255 spanning-disabled
!
interface FastEthernet0.666
 encapsulation dot1Q 666
 no ip route-cache
 bridge-group 254
 no bridge-group 254 source-learning
 bridge-group 254 spanning-disabled
!
interface BVI1
 ip address 10.0.0.1 255.0.0.0
 no ip route-cache
!
ip default-gateway 10.0.0.2
ip http server
no ip http secure-server
ip http help-path http://www.cisco.com/warp/public/779/smbiz/prodconfig/help/eag
ip radius source-interface BVI1
!
access-list 1 permit any
radius-server attribute 32 include-in-access-req format %h
radius-server host 10.0.0.3 auth-port 1812 acct-port 1813 key 7 00171605165E1F
radius-server vsa send accounting
!
control-plane
!
bridge 1 route ip
!
!
!
line con 0
 transport preferred all
 transport output all
line vty 0 4
 transport preferred all
 transport input all
 transport output all
line vty 5 15
 transport preferred all
 transport input all
 transport output all
!
end
```

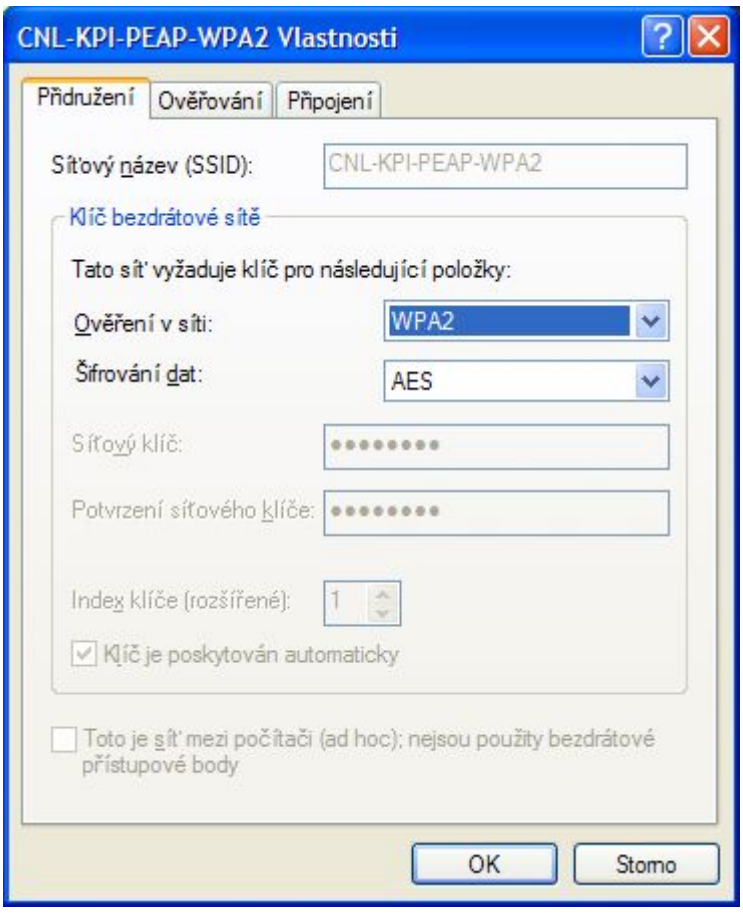
- for more information refer to www.cisco.com

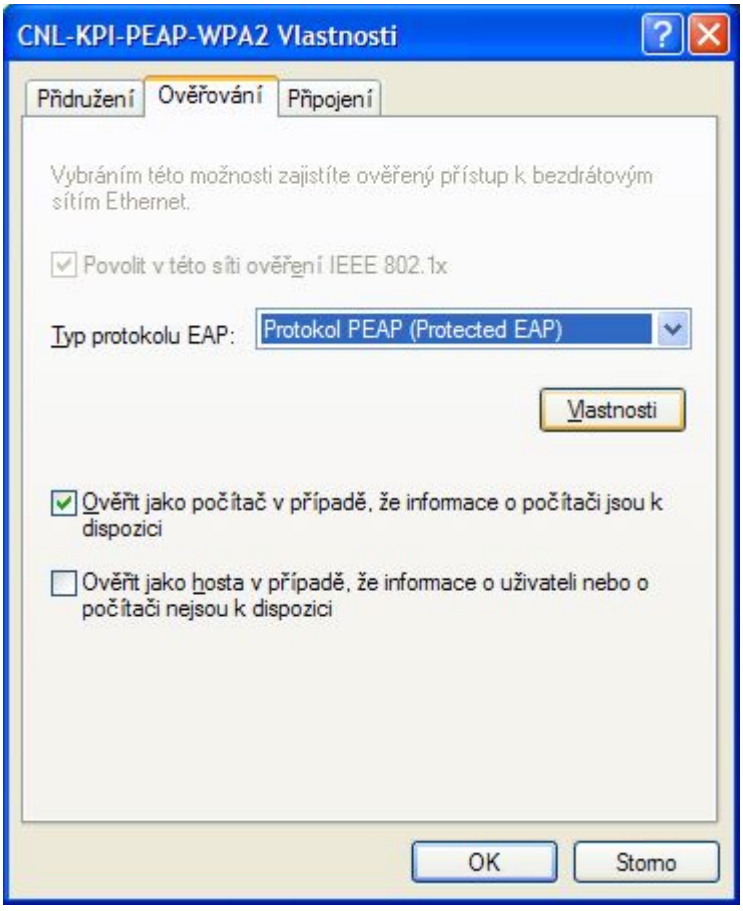
4. Windows XP

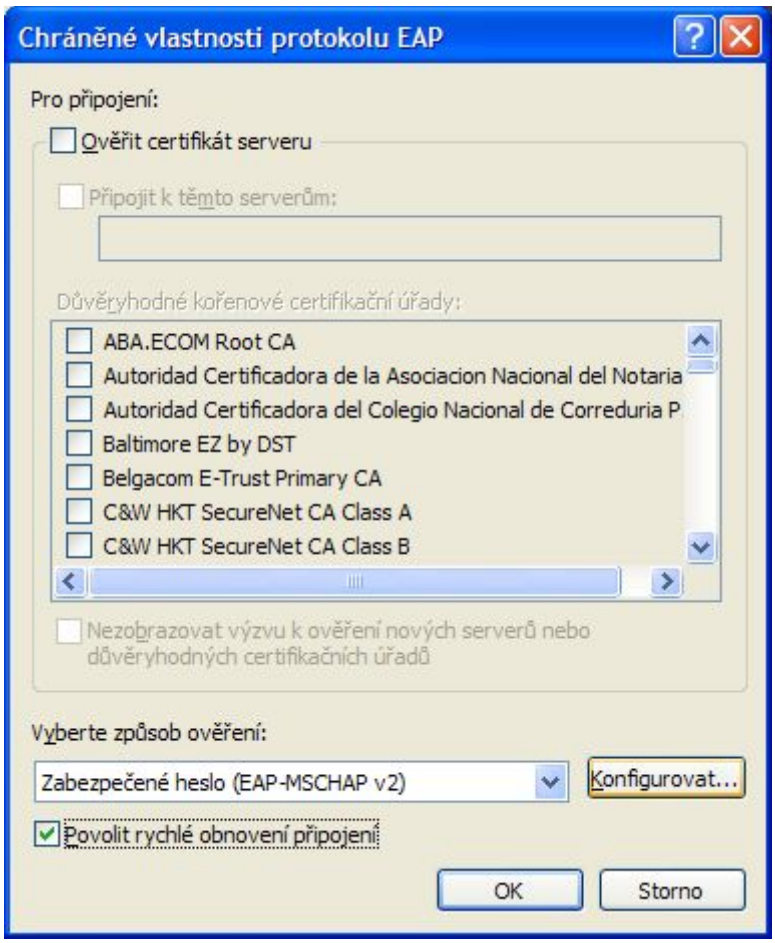
- configure windows as shown in the next screenshots.

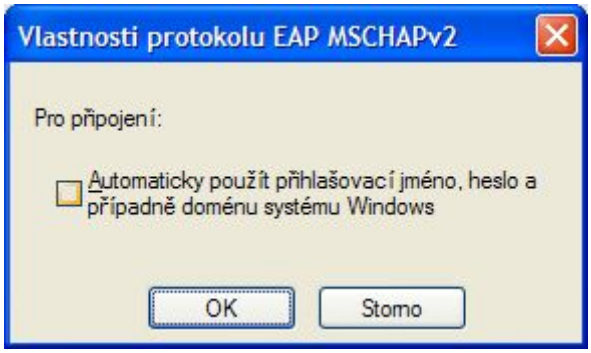




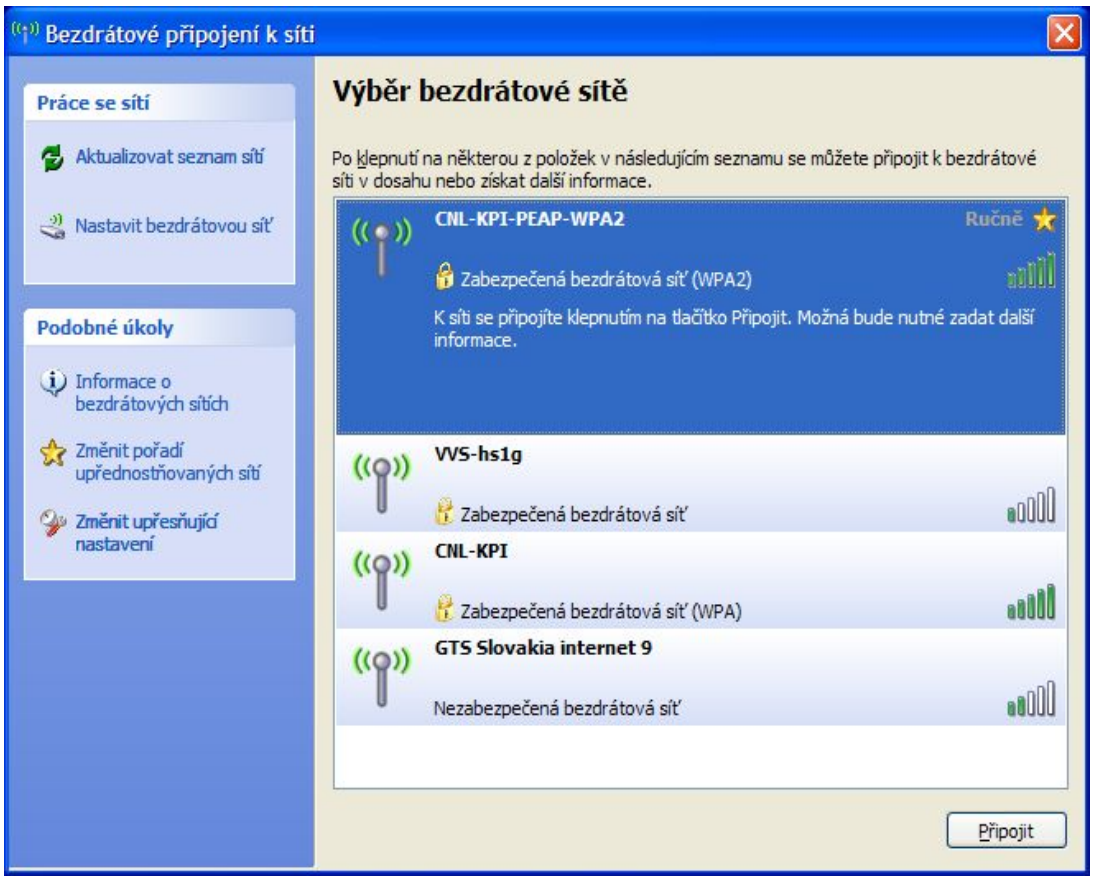

CNL-KPI-PEAP-WPA2 Vlastnosti
Přidružení
Ověřování
Připojení
Síťový název (SSID):
CNL-KPI-PEAP-WPA2
Klíč bezdrátové sítě
Tato síť vyžaduje klíč pro následující položky:
Ověření v síti:
WPA2
Šifrování dat:
AES
Síťový klíč:
Potvrzení síťového klíče:
Index klíče (rozšířené):
1
Klíč je poskytován automaticky
Toto je síť mezi počítači (ad hoc); nejsou použity bezdrátové přístupové body
OK
Storno

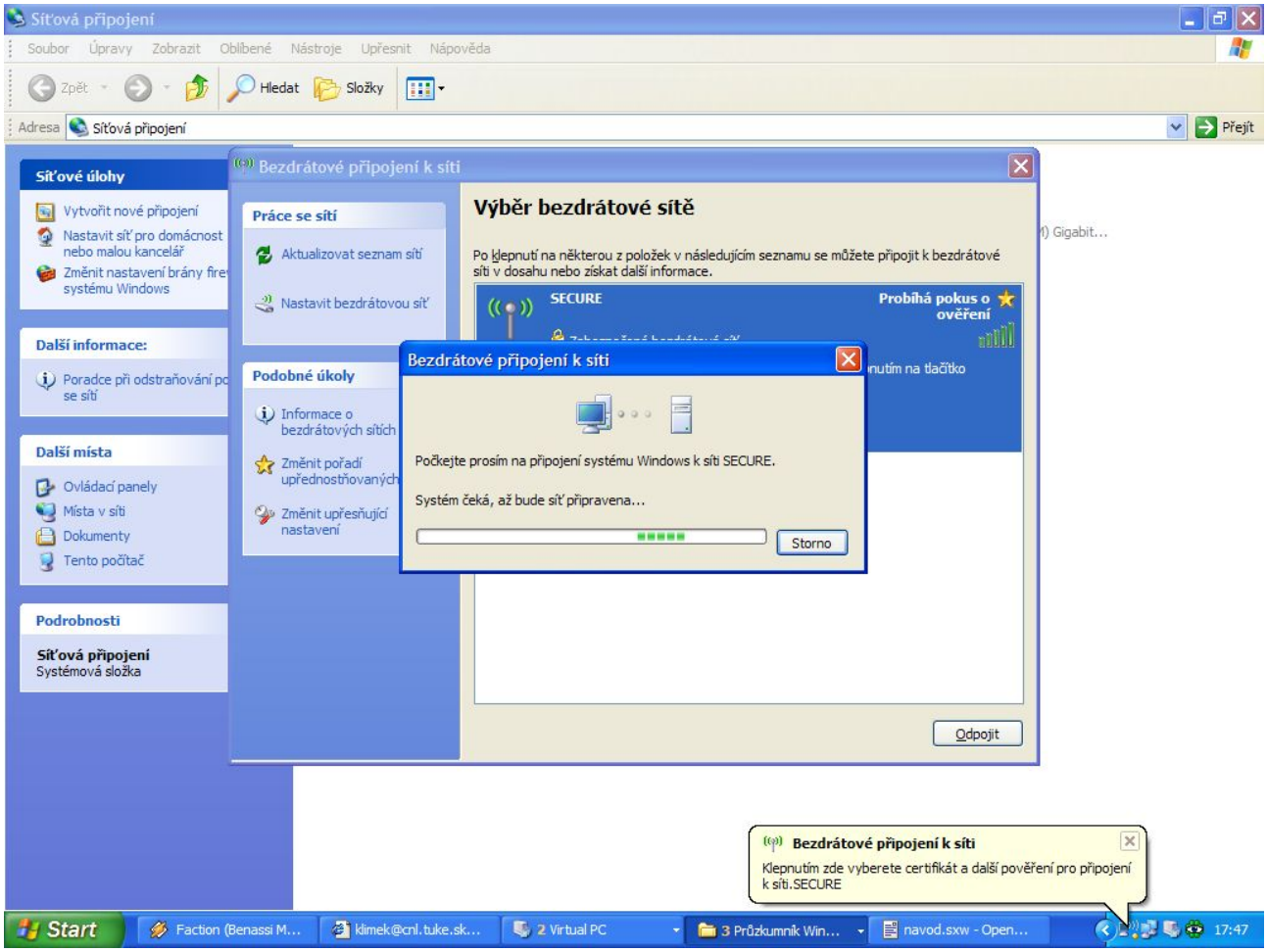

CNL-KPI-PEAP-WPA2 Vlastnosti
Přidružení
Ověřování
Připojení
Vybráním této možnosti zajistíte ověřený přístup k bezdrátovým sítím Ethernet.
Povolit v této síti ověření IEEE 802.1x
Typ protokolu EAP:
Protokol PEAP (Protected EAP)
Vlastnosti
Ověřit jako počítač v případě, že informace o počítači jsou k dispozici
Ověřit jako hosta v případě, že informace o uživateli nebo o počítači nejsou k dispozici
OK
Storno





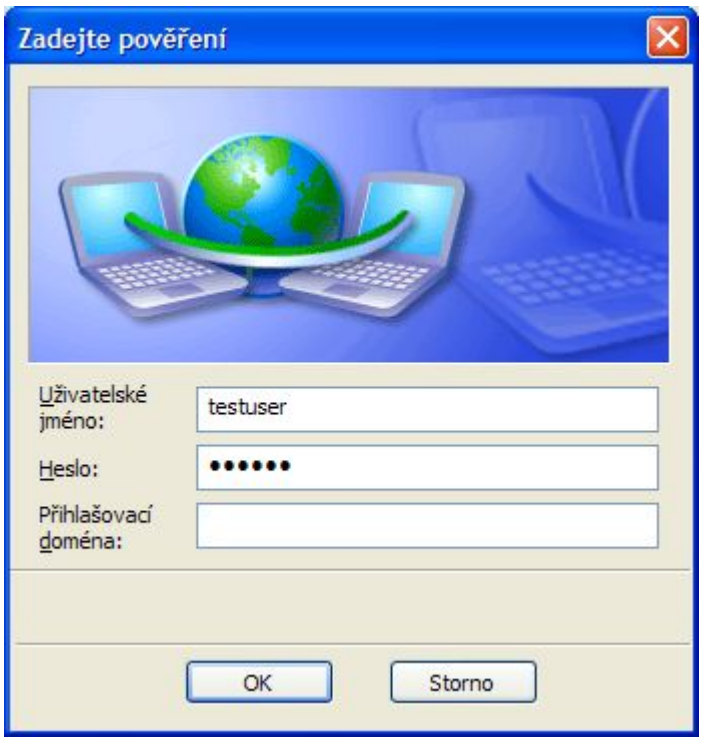
- configuration is finished, now try to connect.



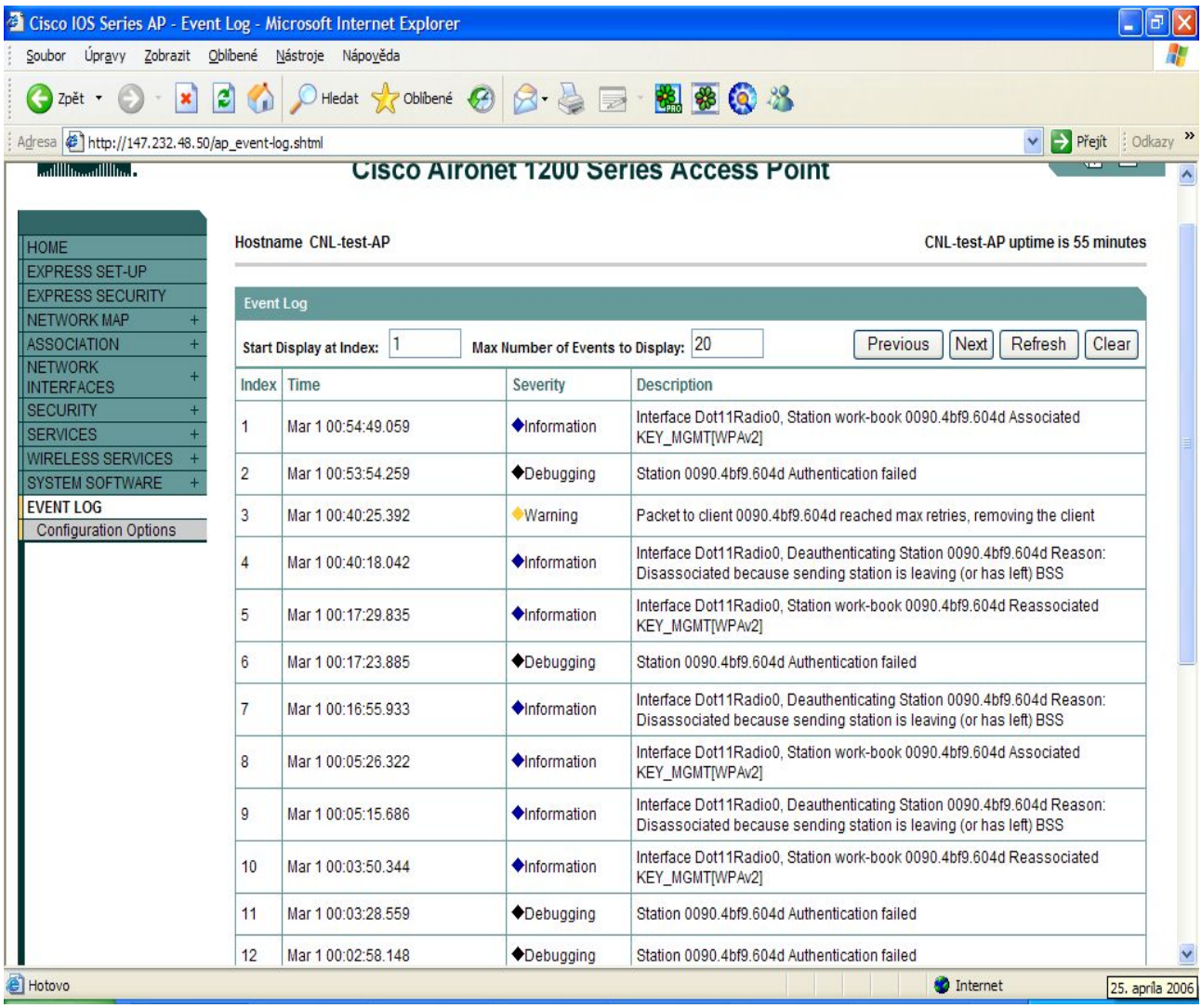


- to configure the username and password click on the bubble in the

right bottom of the screen.



- look at the first line, (with index 1) it says that the user was sucessufly associated, the lines bellow it (index 2,6) show what will happen when the radius daemon isnt running - authentification will fail.



4. Literature

http://tldp.org/HOWTO/html_single/8021X-HOWTO/
http://www.cisco.com
http://www.microsoft.com
http://lists.freeradius.org
http://mattzz.dyndns.org/twiki/bin/view/Projects/FreeRadiusAuthentication
http://lists.cistron.nl
http://vuksan.com/linux/dot1x/802-1x-LDAP.html