Contents

Abbreviations .................................................................................................................................. 3

1 Introduction.......................................................................................................................... 4

2 Used hardware and Software........................................................................................ 5

3 Results............................................................................................................................... 7
  3.1 RADIUS Server ........................................................................................................... 7
  3.2 Access Point............................................................................................................... 7
  3.3 Source code............................................................................................................... 8

4 Configuration ................................................................................................................... 9
  4.1 HostAP daemon ......................................................................................................... 10
  4.2 Linux client ............................................................................................................... 11
  4.3 WindowsXP client .................................................................................................... 11
  4.4 Radiator RADIUS server ....................................................................................... 12
### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAA</td>
<td>Authentication, Authorisation and Accounting</td>
</tr>
<tr>
<td>AP</td>
<td>Access Point</td>
</tr>
<tr>
<td>CHAP</td>
<td>Challenge-Handshake Authentication Protocol</td>
</tr>
<tr>
<td>CPAN</td>
<td>Comprehensive Perl Archive Network</td>
</tr>
<tr>
<td>EAP</td>
<td>Extensible Authentication Protocol</td>
</tr>
<tr>
<td>GPL</td>
<td>General Public License</td>
</tr>
<tr>
<td>IAPP</td>
<td>Inter-Access Point Protocol</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>MD5</td>
<td>Message Digest 5, encryption algorithm</td>
</tr>
<tr>
<td>MS</td>
<td>Microsoft</td>
</tr>
<tr>
<td>PCI</td>
<td>Peripheral Component Interconnect</td>
</tr>
<tr>
<td>PCMCIA</td>
<td>Personal Computer Memory Card International Association</td>
</tr>
<tr>
<td>PEAP</td>
<td>Protected Extensible Authentication Protocol</td>
</tr>
<tr>
<td>RADIUS</td>
<td>Remote Authentication Dial-In User Service</td>
</tr>
<tr>
<td>TLS</td>
<td>Transport Layer Security</td>
</tr>
<tr>
<td>TTLS</td>
<td>Tunneled TLS</td>
</tr>
<tr>
<td>WLAN</td>
<td>Wireless Local Area Network</td>
</tr>
</tbody>
</table>
1 Introduction

In the period of 1 march 2004 until 25 march 2004, a study has taken place on how to build an authentication environment where all elements communicate via IPv6. The purpose of this project was to create an 802.1x authentication environment in an IPv6 environment, based on RADIUS. The study has taken place for the SurfNet Freeband Testbed (http://www.freeband.nl/testbed/).

The goal is to have more insight in issues arising from implementing and the role-out of 802.1x in an IPv6 environment. Besides that, it will be possible to use the newest technologies even before commercial products are available.

The IEEE-standard 802.1x, Port Based Network Access Control, is used a lot these days for securing the WLAN, and is therefore seen in many access points. 802.1x is, like 802.11a and 802.11b, a transport protocol between client terminal and the access point (AP).

In this way, the AP is able to exchange Extensible Authentication Protocol (EAP) messages with the client. EAP provides different means of authentication, such as:

- MD5 (Username/Password)
- MS-CHAPv2 (Microsoft Username/Password)
- PEAP (secure MS-CHAP)
- TLS (certificates)
- TTLS (tunneled TLS)

On the other side, the AP is connected to a (Remote Authentication Dial-In User Service) RADIUS Server. This is where the authentication actually takes place. The AP makes use of the existing IP infrastructure (See Figure 1). Current available commercial APs do not support IPv6. This means that 802.1x cannot be used in an IPv6 environment.

![Figure 1: Overview of the used protocols](image)

The project goal can therefore be translated to the creation of an AP, which can send RADIUS messages over an IPv6 infrastructure.
2 Used hardware and Software

A DELL laptop is used as client. The laptop is equipped with a D-Link DWL-650 wireless LAN PCMCIA card (802.11b). Both WindowsXP professional and Red Hat 9 Linux are used as operating systems.

WindowsXP supports 802.1x and under Linux Open1x xsupplicant is used.


The AP is a PC with Fedora Linux 1.0. The standard kernel has been replace by the 2.4.25 kernel. A PCMCIA-PCI bridge is installed and a D-Link DWL-650 wireless LAN PCMCIA card is installed for the wireless interface (802.11b). The PC is equipped with a standard Ethernet card (3com 3c...)

HostAP version 1.3.0 (http://hostap.epitest.fi/) is installed on the PC. This software is available under GPL. HostAP is a wireless LAN driver, which makes it possible for the PC to act as an AP.

HostAP is a driver for Prism2, -2.5 and -3 chipsets. The driver is IP independent and basically offers the support of WLAN under Linux. HostAP furthermore offers more intelligence via a daemon, e.g. access control based on MAC addresses and/or with the use of 802.1x.

Besides that, HostAP has implemented a first version of Inter Access Point Protocol (IAPP). Using IAPP, APs can exchange information about associated clients. This can make the roaming of client much faster.

The possibilities provided by HostAP as described above makes HostAP more suitable for this project than other solutions such as OpenAP (http://opensource.instant802.com/) and LinuxAP (http://linuxap.ksmith.com/).

The created AP and the client laptop are integrated in the network architecture of the ‘ICT doorbraakproject’ Beyond3G. This is shown in Figure 2.

---

1 WindowsXP is standard for the DELL laptops. Red Hat 9 Linux is used for other projects making use of the same laptop. The D-link wireless LAN card is used because the used chipset (Prism2) is best supported under Linux.

2 Fedora Linux 1.0 was the most recent version of the Red Hat group at this time. The kernel has been replaced to support all used hardware. The 2.4.25 kernel supports all hardware.

3 HostAP is a wireless driver for Prism2, -2.5 en -3 chipsets. This is the reason for using the D-Link wireless LAN card.

4 IAPP is an IPv4 based protocol and cannot be used in an IPv6 environment.
Figure 2: IPv6 network architecture with AAA
3 Results

3.1 RADIUS Server

After applying the latest patches, Radiator supports requests over IPv6. The majority of Radiator is written in Perl and the patches to support IPv6 require the Socket6 module of CPAN (Comprehensive Perl Archive Network).

3.2 Access Point

The HostAP software, which makes the PC act as an AP, was more difficult to get running properly. HostAP consists of three parts:

- HostAP driver; a Linux WLAN driver that supports Ad-hoc and managed mode but also offers the Master mode to act like an AP.
- HostAP daemon; a userspace daemon that offers extra communication and administrator possibilities for the AP.
- HostAP utilities; a number of configuration utilities.

HostAP requires the use of a WLAN card with a Prism2, -2.5 or -3 chipset.

After installing the HostAP software, the WLAN card immediately worked in the Master mode. To support encryption and 802.1x, the daemon must be activated as well. HostAP daemon, however, does not support the use of IPv6. Before porting the daemon to IPv6, a set-up with IPv4 was made. This revealed a bug in HostAP, which was corrected. The bug fix is indicated in the source with the following code:

```
#ifdef __BUGS
    code
#else
#endif
```

After having a working IPv4 set-up, the code of the HostAP daemon has been changed to support IPv6. Because of the short duration of this project, only the necessary changes have been made to be able to send RADIUS messages over IPv6. This mainly concerns the translation of IPv4 data structure to those of IPv6. The code for the IAPP has not been altered because this is beyond the scope of this project. Also, the goal of the project is to have this operating in an IPv6 environment, not a dual-stack environment. Therefore the use of IPv4 is not supported anymore.

Changes made to the source code are indicated with the following:

```
#ifdef __IPV6
    new code
#else
    old code
#endif
```


Changing the code for IAPP is more difficult because IAPP makes use of broadcasts. IPv6 does not make use of broadcast addresses anymore.
This concerns the following files:
- config.h
- config.c
- radius_client.c

With a small change in the makefile, the code can be compiled again. After this, the HostAP daemon is able to send RADIUS messages over IPv6.

3.3 Source code

The source code is available for those who are interested. Please contact Olof Schuring (TNO Telecom) for more information and the code.

email: O.J.Schuring@telecom.tno.nl
Tel: +31 15 28 57260
4 Configuration

To have the authentication procedure fully functional, the following files have to be configured properly:

For the access point:
- hostapd.conf

For the Linux client:
- lx.conf

For the WindowsXP client:
- certificates
- WLAN configuration

For the Radiator RADIUS server:
- radius.cfg
- users-file

These files are shown in paragraph 4.1 till paragraph 4.4.

Note:
The file “/etc/hostapd.conf” uses “own_ip_addr”. Because the source code of HostAP is not fully ported to IPv6, this is still an IPv4 address. This can be an arbitrarily IPv4 address, since this is not used for any communication.

More information about the files, their purpose and configuration possibilities, is given in the original configuration files.

\[ All \text{ sensitive information that is provided in these files has been replaced by } [a\_description]. \]
4.1 HostAP daemon

/etc/hostapd.conf
#Configuration file of the hostap daemon

interface=wlan0
logger_syslog=4
logger_syslog_level=0
logger_stdout=4
logger_stdout_level=1
dump_file=/tmp/hostapd.dump
daemonize=1
ssid=IPv6AP
macaddr_acl=0
deny_mac_file=/etc/hostapd.deny
auth_algs=3
ieee8021x=1
minimal_eap=0
eap_message=hello
wep_key_len_broadcast=5
wep_key_len_unicast=5
wep_rekey_period=300
eapol_key_index_workaround=1
own_ip_addr=10.0.10.1
auth_server_addr=[radius_server_ipv6_addr]
auth_server_port=1812
auth_server_shared_secret=[a_shared_secret]
acct_server_addr=[radius_server_ipv6_addr]
acct_server_port=1813
acct_server_shared_secret=[a_shared_secret]
4.2 Linux client

1x.conf
#configuration file for open1x xsupplicant

IPv6AP : id [User@domain.com]
IPv6AP : cert = /etc/1x/certs/user-cert.cer
IPv6AP : key = /etc/1x/certs/user-key.pem
IPv6AP : root = /etc/1x/certs/cacert.pem
IPv6AP : auth = EAP
IPv6AP : type = wireless
IPv6AP : pref = tls
IPv6AP : chunk_size = 1398
IPv6AP : random_file = /dev/random
IPv6AP : after_auth = "/bin/echo I authenticated"

4.3 WindowsXP client

For WindowsXP, the root and user certificates must be installed. Besides that, 802.1x authentication must be selected for the WLAN configuration. More information can be found at:

4.4 Radiator RADIUS server

/etc/radiator/radius.cfg

LogDir /var/log/radius
DbDir /etc/radiator
AuthPort 1812
AcctPort 1813
Trace 3

# Listen on all IPv6 and IPv4 addresses
BindAddress ipv6:::

# Surfnet connection...
<Client [surfnet_ipv4_address]>
  Secret [surfnet_shared_secret]
</Client>

<Client DEFAULT>
  Secret [a_shared_secret]
  DupInterval 0
</Client>

<Realm domain.com>
  <AuthBy FILE>
    Filename %D/users
    EAPType TLS
    EAPTLS_SessionResumption 0
    EAPTLS_CertificateFile %D/certs/cacert.pem
    EAPTLS_CertificateType PEM
    EAPTLS_PrivateKeyFile %D/certs/server-cert.pem
    EAPTLS_PrivateKeyPassword [password]
    EAPTLS_MaxFragmentSize 1024
    AutoMPPEKeys
  </AuthBy>
  AcctLogFileName %D/detail
</Realm>

# All other traffic to forwarded
<Realm DEFAULT>
  <AuthBy RADIUS>
    Host [surfnet_ipv4_address]
    LocalAddress 0.0.0.0
    Secret [surfnet_shared_secret]
    AuthPort 1812
    AcctPort 1813
    Retries 1
  </AuthBy>
</Realm>
/etc/radiator/users

#User-configuration file for Radiator RADIUS server

[User@domain.com]  Auth-Type := EAP
Tunnel-Type = VLAN,
Tunnel-Medium-Type = Ether_802,
Tunnel-Private-Group-ID = 1