RADIUS

September 2000

RADIUS

Table of Contents

REFERENCE

1 RADIUS

This chapter gives you information on the RADIUS implementation of BinTec Communications AG. You will learn how to configure a BinTec router as a RADIUS client and what is necessary to know about configuring a RADIUS server. Useful examples are given.

The following items are covered:

1.1 Overview

Client / Server

RADIUS (Remote Authentication Dial In User Service) is a client/server protocol originally developed by Livingston Enterprises. RADIUS provides a security system that allows you to exchange authentication and configuration information between a Network Access Server, such as the BRICK, and a RADIUS

Server, a PC or UNIX machine running a RADIUS daemon process. The RADIUS server maintains a database of user authentication data and configuration information.

RADIUS can be used for:

- Authentication
- Accounting

The BinTec router sends a request with username and password to the RADIUS server, the server examines its database. If the user is found and may connect, the RADIUS server returns an accept message to the BinTec router. The message contains parameters (RADIUS attributes) that the BinTec router uses for the configuration and further negotiation of the related WAN connection.

When using a RADIUS server for accounting, the BinTec router sends an accounting start record at the beginning and a stop record at the end of every connection. These start and stop records also contain RADIUS attributes describing the connection (IP address, username, throughput, charges).

RADIUS packets

The following types of packets are sent between RADIUS server and RADIUS client:

Types	Sent from –> to	Purpose
ACCESS_REQUEST	Client –> Server	When a connection request is received on the BRICK the RADIUS server is polled if a locally defined PPP partner could not be found (i.e., upon receiving the calling partner's PPP_ID and no local record exists for the PPP partner.).
ACCESS_ACCEPT	Server –> Client	If the RADIUS server authenticates the information contained in the ACCESS_REQUEST packet, it sends an ACCESS_ACCEPT packet to the RADIUS client that contains the link setup parameters to use.
ACCESS_REJECT	Server –> Client	If the information contained in the ACCESS_REQUEST packet doesn't match the information in the RADIUS Server's user database (usually /etc/raddb/users) the server may deny access to the network.

Types	Sent from –> to	Purpose
ACCOUNTING_START	Client -> Server	When using a RADIUS server for accounting, the BinTec router sends an accounting start record at the beginning of every connection.
ACCOUNTING_STOP	Client -> Server	When using a RADIUS server for accounting, the BinTec router sends an accounting stop record at the end of every connection.

Table A-1: RADIUS packets

Configuration steps

The required configuration steps have to be done on:

- BRICK side (see section A, chapter 1.2, page 8)
- RADIUS server side (see section A, chapter 1.3, page 18)

RADIUS table entries

The ifIndexes of RADIUS PPP entries start at 15001. They are not stored when saving your configuration.

Further information on RADIUS

For further information on RADIUS, here are some useful links:

- A nice little introduction to RADIUS:
  http://www.squashduck.com/~roundman/radius/
- A useful site with lots of information on RADIUS and sources of supply for RADIUS servers:
  http://www.dnt.ro/~vsv/radius.html
- A BinTec FAQ concerning the Steel-Belted RADIUS server and configuring callback for Windows clients:
  http://www.bintec.de/gb/service/index.html

1.2 Configuration on BinTec router Side

The BinTec router can be configured via

- Setup Tool (see section A, chapter 1.2.1, page 9)
- MIB variables (see section A, chapter 1.2.2, page 14)

1.2.1 Setup Tool

The menu ***IP*** ➧ ***RADIUS SERVER*** lists all RADIUS servers currently configured on the router.

```
BinTec router                                        BinTec Communications AG
                                                                     MyRouter

Proto    Prio     IP Address       State

auth     0        111.11.11.11     active




Press <Ctrl-n>,<Ctrl-p> to scroll through menu items,<Return> to enter
```

You can add, edit, or delete list entries in the usual fashion.

The configuration of a RADIUS server is made in ***IP*** ➧ ***RADIUS SERVER*** ➧ ***ADD***:

```
BinTec router                                        BinTec Communications AG
[IP][RADIUS][ADD]:Configure Radius Server                             MyRouter

Protocol                 auth

IP Address               44.55.66.77
Password                 blubb

Priority                 0
Policy                   authoritative

Port                     1812
Timeout                  1000
Retries                  1
State                    active

                 SAVE

Use <Space> to select
```

The menu contains the following entries:

Field	Meaning
Protocol	Defines whether the RADIUS server is used for authentication purposes or for accounting ISDN connections. Possible values: ■ *auth* (default value): Authentication. ■ *acct:* Accounting.
IP Address	The IP address of the RADIUS server.
Password	This is a shared secret between RADIUS server and BinTec router.
Priority	Priority of the RADIUS server. When there are several RADIUS server entries, the server with the highest priority is used first. If there is no reply from this server, the server with the next highest priority is used and so forth. Possible values: Integers from *0* (highest priority) to *7* (lowest priority). Default value: *0*.
Policy	Defines how the BinTec router reacts when receiving a negative answer to a request. Possible values: ■ *authoritative* (default value): A negative answer to a request will be accepted. ■ *non-authoritative*: A negative request will not be accepted, but the next RADIUS server will be asked until there is finally an authoritative server configured.

Field	Meaning
Port	Number of TCP port to use for RADIUS data. According to RFC 2138 the default ports are 1812 for authentication (was 1645 in older RFCs) and 1813 for accounting (was 1645 in older RFCs). Many RADIUS servers, including Merit, still use 1645 and 1646. As RADIUS servers use different port numbers, you should refer to the documentation for the RADIUS server you are using. Default value: *1812*.
Timeout	Number of milliseconds to wait for an answer to a request. Possible values: Integers from *50* to *50000*. Default value: *1000* (1 second).
Retries	Number of retries if a request is not answered. If, after these attempts, still no answer has been received, the server **State** is set to *inactive*. The BinTec router then tries to contact the server every 20 seconds, and once the server replies, **State** is changed to *active* again. Possible values: Integers from *0* to *10*. Default value: *1*. To prevent the **State** switching to *inactive*, set this value to *0*.
State	The state of the RADIUS server. Possible values: ■ *active* (default value): Server answers requests. ■ *inactive*: Server does not answer (see **Retries** above). ■ *disabled*: Requests to a certain RADIUS server are temporarily disabled.

Table A-2: ***IP*** ➧ ***RADIUS SERVER*** ➧ ***ADD***

Menu *PPP* For incoming calls there are some options that can not be set user specific. They have an effect on the PPP negotiation and RADIUS server usage before the caller can be identified by username and password. These settings are entered in the menu ***PPP***:

```
BinTec router Setup Tool                        BinTec Communications AG
[PPP]:PPP Profile Configuration                                 MyRouter


    Authentication Protocol           CHAP + PAP + MS-CHAP
    Radius Server Authentication      inband

    PPP Link Quality Monitoring       no


            SAVE                                  CANCEL

Use <Space> to select
```

PPP contains the following items:

Field	Meaning
Authentication Protocol	Defines the PPP authentication protocol offered to the caller first.
Radius Server Authentication	Is used to configure possible RADIUS authentication on incoming calls. Possible values: ■ *inband* (default value): Only inband RADIUS requests (PAP,CHAP) are sent to the specified RADIUS server. ■ *Calling Line Identification (CLID):* Only outband requests are sent to the RADIUS server. ■ *CLID + inband*: Both requests are sent to the RADIUS server (first outband request, then inband request if necessary). ■ *none*: No requests are sent.
PPP Link Quality Monitoring	Defines whether Link Quality Monitoring is executed for PPP connections.

Table A-3: ***PPP***

1.2.2 MIB

RadiusServerTable

Configuration is made over **RadiusServerTable**, it contains the following variables:

Variable	Meaning
RadiusSrvProtocol	Defines whether the RADIUS server is used for authentication purposes or for accounting ISDN connections. Possible values: ■ *authentication* (default value). ■ *accounting.*
RadiusSrvAddress	The IP address of the RADIUS server.
RadiusSrvPort	Number of TCP port to use for RADIUS data. According to RFC 2138 the default ports are 1812 for authentication (was 1645 in older RFCs) and 1813 for accounting (was 1645 in older RFCs). Many RADIUS servers, including Merit, still use 1645 and 1646. As RADIUS servers use different port numbers, you should refer to the documentation for the RADIUS server you are using. Default value: *1812*.
RadiusSrvSecret	This is a shared secret between RADIUS server and BinTec router.
RadiusSrvPriority	Priority of the RADIUS server. When there are several RADIUS server entries, the server with the highest priority is used first. If there is no reply from this server, the server with the next highest priority is used and so forth. Possible values: Integers from *0* (highest priority) to *7* (lowest priority). Default value: *0*.

Variable	Meaning
RadiusSrvTimeout	Number of milliseconds to wait for an answer to a request. Possible values: Integers from *50* to *50000*. Default value: *1000* (1 second).
RadiusSrvRetries	Number of retries if a request is not answered. If after these attempts still no answer was received, the **RadiusSrvState** is set to *inactive*. The BinTec router then tries to contact the server every 20 seconds, and once the server replies, **RadiusSrvState** is changed to *active* again. Possible values: Integers from *0* to *10*. Default value: *1*. To prevent the **RadiusSrvState** switching to *inactive*, set this value to *0*.
RadiusSrvState	The state of the RADIUS server. Possible values: ■ *active* (default value): Server answers requests. ■ *inactive*: Server does not answer (see **RadiusSrvRetries** above). ■ *disabled*: Requests to a certain RADIUS server are temporarily disabled. ■ *delete*: Deletes the entry.

Variable	Meaning
RadiusSrvPolicy	Defines how the BinTec router reacts when receiving a negative answer to a request. Possible values: ■ *authoritative* (default value): A negative answer to a request will be accepted. ■ *non_authoritative*: A negative request will not be accepted, but the next RADIUS server will be asked until there is finally an authoritative server configured.
RadiusSrvValidate	This additional option is only used for Bogus RADIUS servers, which send response messages with a miscalculated MD5 checksum. All messages generated by the BinTec router, however, will always use the proper authentication scheme. Possible values: ■ *enabled* (default value). ■ *disabled*. For security reasons this option should always be set to enabled.
RadiusSrvDialout	This option provides the means for RADIUS dial-out configuration. Possible entries: ■ *disabled* (default value). ■ *enabled*: Enables initial loading of dial-out routes after reboot. ■ *reload*: Reload of dial-out routes. For further information about RADIUS for Dial-Out, see section A, chapter 1.6, page 34).

Variable	Meaning
RadiusSrvDefaultPW	Is not required with certain RADIUS implementations, such as Merit. Here you should consult the documentation for your RADIUS server. This is the default USER-PASSWORD the BinTec router sends where no password is available (for example, in requests for the calling number or boot requests). Some RADIUS servers rely on a configured USER or CHAP-PASSWORD for any RADIUS request. The default value is an empty string.

Table A-4: RadiusServerTable

biboPPPProfileTable For incoming calls there are some options that can not be set user specific. They have an effect on the PPP negotiation and RADIUS server usage before

the caller can be identified by username and password. These settings are entered in the **biboPPPProfileTable**:

Variable	Meaning
biboPPPProfileName	The name of the PPP profile.
biboPPPProfileAuthProtocol	The type of authentication used on the point-to-point link as described in RFC 1334.
biboPPPProfileAuthRadius	This entry is used to configure RADIUS authentication on incoming calls. Possible values: ■ *none*: RADIUS requests are not sent to the specified RADIUS server. ■ *inband* (default value): Only inband RADIUS requests (PAP, CHAP) are sent to the specified RADIUS server. ■ *outband*: Only outband RADIUS requests are sent to the server. ■ *both*: Both requests are sent to the RADIUS server (first outband request, then inband request if necessary).
biboPPPProfileLQMonitoring	This parameter enables or disables PPP Link Quality Monitoring (LQM) according to RFC 1989. Only relevant for inband authentication.

Table A-5: **biboPPPProfileTable**

1.3 Configuration on the RADIUS Server

RADIUS server files When configuring a RADIUS server, different files have to be edited:

The files described in the following table are available when using a RADIUS server under Unix.

Using a RADIUS server under Windows, the configuration takes place in another way, but it is the same principle.

File	default location	Remarks
radiusd	/etc/raddb/	The RADIUS daemon on UNIX systems.
dictionary	/etc/raddb/	The dictionary file lists the RADIUS attributes the daemon process supports and defines each attribute's default behavior.
clients	/etc/raddb/	The clients file defines the list of hosts that are allowed to request authentication information from the server. Each entry typically contains the RADIUS client's host name and password, (also called the Client-Key).
users	/etc/raddb/	The users file contains user-authentication information for (dial-in) hosts that will be establishing connections via the RADIUS clients. The file consists of user-profiles (also referred to as authentication-lines) that: 1. define requirements for authenticating callers (password, PPP ID, Calling Line) and, 2. define the type of connections to establish if the user has been successfully authenticated.
logfile	/etc/raddb/	The logfile contains error messages from the radiusd process on Unix hosts.
detail	/usr/adm/radacct/<client>/	The detail file contains RADIUS accounting information records submitted by RADIUS clients. <*client*> in the pathname to this file is usually the host name of the RADIUS client.

Table A-6: RADIUS server files

Configuration steps on the RADIUS server

The following steps have to be performed:

- The dictionary file has to be imported. It is available from BinTec's WWW server at http://www.bintec.de (Section: Download). To reach the section Download, click Solutions & Products.
  A list of tested RADIUS servers (eventually with an adapted dictionary file) is also available from BinTec's WWW server at http://www.bintec.de/de/prod/index.html (Section: Lösungen für unterschiedliche Unternehmensgrößen).
  For further information concerning the syntax of dictionary files of definite RADIUS servers, see section A, chapter , page 29.
- The BinTec router has to be entered as NAS (Network Access Server) server and the shared secret has to be entered (Clients file under Unix).
- The correct port has to be entered (as RADIUS servers use different port numbers, you should refer to the documentation for the RADIUS server you are using).
- The users have to be entered (users file under Unix). Here you can define for each user:
  – authentication information (username, password)
  – configuration information which is transferred from the RADIUS server to the RADIUS client (e. g. IP address, callback, access lists, etc.)

  Therefore, you use the standard attributes supported by BinTec's RADIUS implementation and BinTec extensions (see section A, chapter 1.4, page 20).

If you use a RADIUS server for accounting, be sure to have a strategy for packing, moving and accounting files!

1.4 Authentication

To use the RADIUS server for the purpose of authentication, you can define several attributes for each user. The RADIUS server transfers this configuration information to the RADIUS client when accepting the authentication.

In the tables below, all supported standard RADIUS attributes (see section A, chapter 1.4.1, page 21) and BinTec extensions (see section A, chapter 1.4.2, page 27) are listed.

The following standard attributes were implemented to fully conform with RFC 2058 (Remote Authentication Dial In User Service RADIUS).

The values of the attributes can have the following types:

Value	Meaning
string	0 - 253 octets
integer	32-bit value in big endian order (high byte first)
ipaddrs	4 octets in network byte order

Table A-7: Values for Type

1.4.1 List of Standard Attributes Supported

Your router supports the following standard RADIUS attributes. Also a couple of BinTec-specific options have been added to facilitate using your router in conjunction with RADIUS servers.

Note, however, that the BinTec-specific options are only available if you use the dictionary file (available from BinTec's WWW server).

RADIUS attribute	No.	Type	Corresponding MIB variable	R / A	Remarks
User-Name	1	string	**biboPPPAuthIdent** or **biboDialNumber**	REQ	User name, mandatory. Values: ■ inband: PPP partner name. ■ outband: PPP partner telephone number. If outband authentication (CLID) is requested, configuration in **pppProfileTable** has to be done (see section A, chapter 1.7.2, page 43).
User-Password	2	string		REQ	Password for PAP authentication. In case of outband authentication, a password is not available. If your RADIUS server requires a password, set **RadiusSrvDefaultPW** in **RadiusServerTable**.
CHAP-Password	3	string		REQ	Password for CHAP authentication.
NAS-Port	5	integer		ANS	Corresponds to the ISDN stack used for the connection.
Service-Type	6	integer		ANS	Values: ■ for PPP: Framed (2). ■ for PPP callback (CBCP) or Microsoft callback: Callback-Framed (4).

RADIUS attribute	No.	Type	Corresponding MIB variable	R / A	Remarks
Framed-Protocol	7	integer		ANS	Modifications only take affect in case of outband authentication (CLID) or in case of RADIUS used for dial-out. Values: see table A-9, page 26.
Framed-IP-Address	8	ipaddr	**biboPPPIpAddress**	ANS	Partner IP address. Note: With Framed-IP-Address = 255.255.255.254 an IP address from an IP address pool on the BinTec router is assigned (dynamic server mode).
Framed-IP-Netmask	9	ipaddr		ANS	Partner IP netmask.
Framed-Routing	10	integer	**ipExtIfRipSend**	ANS	Defines which entries in the **ipExtIfTable** are set. Values: ■ None (0): No entry. ■ RIPv1-Broadcast(1): **ipExtIfRipSend** gets the value *ripV1*. ■ RIPv1-Listen(2): **IpExtRipReceive** gets the value *ripV1*. ■ RIPv1-Broadcast-Listen (3): **ipExtIfRipSend** gets the value *ripV1* and **IpExtRipReceive** gets the value *ripV1*.
Filter-Id	11	string	**ipExtIfRuleIndex**	ANS	**ipExtIfRuleIndex** is set to <Filter-Id>.

RADIUS attribute	No.	Type	Corresponding MIB variable	R / A	Remarks
Framed-MTU	12	integer		ANS	Is replaced by MRU/MRRU.
Framed-Compression	13	integer		ANS	Compression. Values: ■ None (0) ■ Van-Jacobson-TCP-IP (1)
Reply-Message	18	string	**ifDescr**	ANS	Outband: interface name (**ifDescr**) is set to this name, instead of using the telephone number.
Callback-Number	19	string	**biboDialTable**	ANS	Telephone number for callback. An entry in **biboDialTable** is created.
Framed-Route	22	string		ANS	You can create a routing entry (see table A-10, page 27).
Framed-IPX-Network	23	integer		ANS	If not ffffffe: Where necessary, entries in **ipxCircTable** (**ipxCircType** can get the value *wanRIP* or *unnumberedRIP*), **ripCircTable** and **sapCircTable** are made.
Class	25	string		ANS	If returned by RADIUS server, this attribute is added to every RADIUS accounting record.
Vendor-Specific	26	string		ANS	Only for encapsulation.
Session-Timeout	27	integer		ANS	Not used!
Idle-Timeout	28	integer	**biboPPPShortHold**	ANS	Shorthold.
Called-Station-Id	30	string		REQ	Called phone number.
Calling-Station-Id	31	string		REQ	Calling phone number (is often empty for analog users).

RADIUS attribute	No.	Type	Corresponding MIB variable	R / A	Remarks
NAS-Identifier	32	string	**biboAdmSysName**	REQ	System Name of the BinTec router.
CHAP-Challenge	60			REQ	Necessary for CHAP.
NAS-PORT-TYPE	61				This attribute indicates the type of the physical port of the NAS which is authenticating the user. It is used in addition to the NAS-Port (5) attribute.
Port-Limit	62	integer	**biboPPPMaxConn**	ANS	Number of B channels that are allowed for this user.

Table A-8: Standard RADIUS attributes for authentication

Values for attribute Framed-Protocol

Value	Name	Remarks
1	PPP	inband
17825794	X25	outband
17825795	X25-PPP	
17825796	IP-LAPB	
17825798	IP-HDLC	
17825799	MPR-LAPB	
17825800	MPR-HDLC	
17825801	FRAME-RELAY	
17825802	X31-BCHAN	
17825803	X75-PPP	
17825804	X75BTX-PPP	
17825805	X25-NOSIG	
17825806	X25-PPP-OPT	

Table A-9: Possible values for attribute Framed-Protocol

Values for attribute Framed-Route

With Framed-Route you can create a route of the format:
`Framed-Route = <destaddr/mask> <gateway> <metric>`

The following values are available:

Name	Remarks
destaddr/mask	Destination address with netmask (required for a dial-out request).
gateway	Gateway address (nexthop) (optional).
metric1 - 5	Sets the variables **ipRouteMetric1** to **ipRouteMetric5** in the **ipRouteTable**; metric1 should always lie above the value for the dial-in case, i.e. the worse metric. If no metric is given, metric1 is set to 5, while metric2 to metric5 are set to 0 (optional).

Table A-10: Possible values for attribute Framed-Route

1.4.2 List of BinTec Attributes (Extensions)

If you use the dictionary file mentioned above, you can directly access and configure specific MIB tables via RADIUS.

The syntax of these extensions can change depending on the used RADIUS server. So refer to the documentation of your RADIUS server. For an example see section A, chapter , page 29.

The following BinTec extensions are available at the moment:

Option	No.	Type	Corresponding MIB variable	Mode
BinTec-biboPPPTable	224	string	**biboPPPTable**	static
BinTec-biboDialTable	225	string	**biboDialTable**	dynamic
BinTec-ipExtIfTable	226	string	**ipExtIfTable**	static
BinTec-ipRouteTable	227	string	**ipRouteTable**	dynamic
BinTec-ipExtRtTable	228	string	**ipExtRtTable**	dynamic
BinTec-ipNatPresetTable	229	string	**ipNatPresetTable**	dynamic
BinTec-ipxCircTable	230	string	**ipxCircTable**	dynamic
BinTec-ripCircTable	231	string	**ripCircTable**	dynamic
BinTec-sapCircTable	232	string	**sapCircTable**	dynamic
BinTec-ipxStaticRoute-Table	233	string	**ipxStaticRouteTable**	static
BinTec-ipxStatic-ServTable	234	string	**ipxStaticServTable**	static

Table A-11: BinTec RADIUS extensions

Syntax Each of these options corresponds to a MIB table. You can modify values inside the table by using a syntax similar to the SNMP client shell of your BRICK:

```
<BinTec-Option> = "<variable1>=<value1> … <variablen>=<valuen>"
```

A few lines from a RADIUS users file might look like this:

```
Service-Type = Framed,
BinTec-biboPPPTable = "DynShorthold=50 IpAddress=static",
BinTec-ipNatPresetTable = "Protocol=tcp extport=1050
                           intport=100"
```

When using these options, please note:

- The **ifIndex** is automatically set for each table, you cannot influence it. There is, however, one exception to this rule: In the **IpExtRtTable** both the **DstIfIndex** and the **SrcIfIndex** are automatically set. You can set one of these to *0* if need be.

- The entries are not case-sensitive.
- You must not use blank spaces before or after »=« signs inside the double quotes.
- There are two different option modes, static, and dynamic.
  Static options modify existing table entries while dynamic options add a new table entry. Therefore, all the variables you want to set in a dynamic option have to be included in one single line.

Sample Modification for Merit RADIUS Servers

Merit Here we will give you an example of what the dictionary file on a Merit RADIUS server can look like (Merit 3.6 and later).

The syntax is as follows:

```
<vendor-name>.<vendor-string> <vendor-specific-value> <attribute-number> <attribute-type> <expression>
```

The dictionary file looks like this:

Dictionary file

```
BinTec.attr BinTec-biboPPPTable         224 string(*, 0, NOENCAPS)
BinTec.attr BinTec-biboDialTable        225 string(*, 0, NOENCAPS)
BinTec.attr BinTec-ipExtIfTable         226 string(*, 0, NOENCAPS)
BinTec.attr BinTec-ipRouteTable         227 string(*, 0, NOENCAPS)
BinTec.attr BinTec-ipExtRtTable         228 string(*, 0, NOENCAPS)
BinTec.attr BinTec-ipNatPresetTable     229 string(*, 0, NOENCAPS)
BinTec.attr BinTec-ipxCircTable         230 string(*, 0, NOENCAPS)
BinTec.attr BinTec-ripCircTable         231 string(*, 0, NOENCAPS)
BinTec.attr BinTec-sapCircTable         232 string(*, 0, NOENCAPS)
BinTec.attr BinTec-ipxStaticRouteTable 233 string(*, 0, NOENCAPS)
BinTec.attr BinTec-ipxStaticServTable   234 string(*, 0, NOENCAPS)
```

1.4.3 Microsoft-Specific Attributes Supported

MS-Chap V1/V2 Supported

Provided the RADIUS server supports the authentication protocol and the following attributes, MS-CHAP V1/V2 authentication of PPP dial-in partners is possible:

- MS_CHAP_RESPONSE (authenticate request)

- MS_CHAP2_RESPONSE (authenticate request)

- MS_CHAP_CHALLENGE (authenticate request)

- MS_CHAP2_SUCCESS (authenticate response)

MPPE Encryption Supported

Provided the RADIUS server supports the authentication protocol and the following attributes, MPPE encryption between PPP dial-in partners is possible:

- MS_CHAP_RESPONSE (authenticate request)

- MS_CHAP2_RESPONSE (authenticate request)

- MS_CHAP_CHALLENGE (authenticate request)

- MS_CHAP2_SUCCESS (authenticate response)

- MS_CHAP_MPPE_KEYS (authenticate response)

1.5 Accounting

When you configure a RADIUS server for the purpose of accounting, the BinTec router transmits Start and Stop RADIUS packets for each ISDN connection to this server.

The following standard attributes were implemented to fully conform with RFC 2059 (RADIUS Accounting).

The values of the attributes can have the following types:

Value	Meaning
string	0 - 253 octets
integer	32-bit value in big endian order (high byte first)
ipaddrs	4 octets in network byte order

Table A-12: Values for Type

1.5.1 List of Sent Attributes Supported

The following attributes are available for accounting:

RADIUS attribute	No.	Type	Corresponding MIB variable	Start	Stop	Remarks
User-Name	1	string	**biboPPPAuthIdent** or **biboDialNumber**	X	X	User name.
NAS-Port	5	integer	**isdnCallStkNumber**	X	X	Corresponds to the ISDN stack used for the connection.
Service-Type	6	integer		X	X	The value is always Framed (2).

RADIUS attribute	No.	Type	Corresponding MIB variable	Start	Stop	Remarks
Framed-Protocol	7	integer		X	X	The used encapsulation is always specified as PPP (1).
Framed-IP-Address	8	ipaddr		X	X	This indicates the IP address assigned to the partner that is dialing in.
Class	25	string		X	X	Allows the adjustment of accounting and authentication. If this attribute is sent back from the RADIUS server during authentication, it is inserted to accounting records (depends on the RADIUS server used).
Called-Station-Id	30	string		X	X	Called phone number
Calling-Station-Id	31	string		X	X	Calling phone number (is often empty for analog users).
NAS-Identifier	32	string	**biboAdmSysName**	X	X	Name of the BinTec router.
Acct-Status-Type	40	string		X	X	Possible values: Start, Stop.
Acct-Delay-Time	41	integer		X	X	Time offset in seconds from establishing the connection and sending the accounting record.
Acct-Input-Octets	42	integer			X	Received bytes.
Acct-Output-Octets	43	integer			X	Sent bytes.

RADIUS attribute	No.	Type	Corresponding MIB variable	Start	Stop	Remarks
Acct-Session-Id	44	string		X	X	Common index for multi-link connections, e. g. a2000002.
Acct-Authentic	45	integer		X	X	Indicates how the user was authenticated: by RADIUS, locally or by another remote authentication protocol.
Acct-Session-Time	46	integer			X	Duration of session in seconds.
Acct-Input-Packets	47	integer	**isdnCallReceived-Packets**		X	Received packets.
Acct-Output-Packets	48	integer	**isdnCallTransmit-Packets**		X	Sent packets.
Acct-Terminate-Cause	49	integer			X	This attribute indicates how the session was terminated, and can only be present in Accounting-Request records where the Acct-Status-Type is set to Stop.
Acct-Multi-Session-Id	50	string		X	X	Unambiguous name of the session, e. g. a2000002 (with every call the last 6 digits are incremented).
Acct-Link-Count	51	integer	**biboPPPConnActive**	X	X	Number of B channels, that are established for the connection at the moment.

RADIUS attribute	No.	Type	Corresponding MIB variable	Start	Stop	Remarks
Acct-Charge	59	integer	**isdnCallCharge**		X	Charging units. If this information is not sent as units but as currency amounts (e. g. 0.12 DM), the values are converted to digits (e. g. 120).
NAS-Port-Type	61			X	X	This attribute indicates the type of the physical port of the NAS which is authenticating the user. It is used in addition to the NAS-Port (5) attribute.

Table A-13: Standard RADIUS attributes for accounting

1.6 RADIUS for Dial-Out

As the name suggests (Remote Authentication Dial-In User Service), RADIUS was designed as a client-server system for authenticating dial-in connections. The BinTec router can be configured to operate as a RADIUS client that consults the RADIUS server at connection time for the authentication and identification of specified dial-in partners.

With BinTec's RADIUS implementation it is possible, however, for the BinTec router to request user data from the server in order to establish an PPP connection also for outgoing calls.

Why RADIUS for dial-out

The principal objectives that lay behind the implementation of RADIUS for dial-out are two-fold:

- Firstly, in view of the fact that at most 500 WAN partners can be configured on the BinTec router and some installations can greatly exceed this figure, this feature provides an alternative to configuring WAN partners on the

router. The entries for WAN partners are no longer made locally on the BinTec router via Setup Tool, but now on the RADIUS server. There can thus be considerable savings in terms of Flash memory.

- Secondly, RADIUS for dial-out is easier to manage in terms of configuration. The many entries over Setup Tool are replaced by the more convenient administration of the RADIUS server over the usual editor tools.

How does it work?

The BinTec router firstly requests all the routing information contained on the RADIUS server and stores it in the **ipRouteTable**. Loading of this initial information is driven over the **RadiusSrvDialout** variable. On the one hand, the variable can be set to *enabled* and then saved with the configuration so that initial loading occurs immediately after every reboot. Alternatively, by setting to *reload*, it is possible to load or reload the routing information at any time you choose.

When a dial-out call to a WAN partner is to be made on one of these loaded routes, another request is sent to the RADIUS server in order to receive the necessary partner-specific information (e. g. data for the authentication, encapsulation, extension number etc.), each partner can have more than just one entry in the **ipRouteTable**. If the partner is configured on the RADIUS server, the necessary information entries are transferred to the BinTec router and generated in the respective MIB tables for the duration of the call.

After the end of the call, all entries on the BinTec router´s MIB tables are deleted with the exception of the routing information in the **ipRouteTable**, which is loaded initially.

Configuration

Configuration of RADIUS for dial-out takes place on two levels (similar to configuration for RADIUS for dial-in):

- Configuration on the BRICK side: entries are made over the **RadiusServerTable** (see section A, chapter 1.6.1, page 36).
- Configuration on the RADIUS server: configuration on the users file and dictionary file on the RADIUS server (see section A, chapter 1.6.2, page 36).

1.6.1 Configuration on the BRICK

Configuration on the BinTec router is made over the **RadiusServerTable**.

The required MIB variables are described in section A, chapter 1.2.2, page 14.

1.6.2 Configuration on the RADIUS Server

The following description is taken from a Unix RADIUS implementation, e. g. Merit.

The configuration of the RADIUS server deals with

- telling the users file on the RADIUS server
  - the routing information required for the WAN partner (see "IP routing information", page 37).
  - the partner-specific information assigned to each routing entry (see "Partner-specific information", page 41).
- making sure, that the BinTec-specific extensions are included in the dictionary file of the server.

A significant advantage of the implementation from BinTec Communications AG is that only one entry in the users file is sufficient to enable both dial-in as well as dial-out.

IP routing information

Syntax

This part deals with defining the necessary IP routing information. Here the following syntax should be obeyed:

```
Framed-Route = <destaddr/mask> <gateway> <userid> <userpw>
<private> <metric>
```

Routing info	Meaning
destaddr/mask	Destination address with netmask (required for a dial-out request).
gateway	Gateway address (nexthop) (optional).
userid	For RADIUS for dial-out only. The partner's user ID, necessary for a dial-out request.
userpw	For RADIUS for dial-out only. This entry must match the password attribute in the users file, see section A, chapter 1.7.7, page 46. You need only make the one entry for both dial-in and dial-out. It may not consist only of digits.
private	For RADIUS for dial-out only. Selection of the routing protocols, RIP, OSPF, or the PROXYARP used for the propagation of this IP route.
metric1 - 5	Sets the variables **ipRouteMetric1** to **ipRouteMetric5** in the **ipRouteTable**; metric1 should always lie above the value for the dial-in case, i.e. the worse metric. If no metric is given, metric1 is set to 5, while metric2 to metric5 are set to 0 (optional).

Table A-14: Possible values for attribute Framed-Route

Minimum entries

If only dial-out (without callback) is being configured, destaddr and userid are the minimum entries required.

User dialout-X

Several of these routes are then compiled and arranged under a fictitious user "dialout-X", which begins with the number 1. The number of entries under one of these dummy-users is restricted to the UDP limit of 4096 bytes. Whereby the optimum numbers in terms of loading times and system utilization lie at around 20-40 entries inside the "Framed-Route" record. On initial loading, the BRICK asks for a user by the name of dialout-1, then dialout-2 and so on. Here is an example of what a dummy user could look like:

```
dialout-1
   Framed-Route = "1.2.1.1 user1 secret1 3",
   Framed-Route = "1.2.1.2 user2 secret2 3",
   Framed-Route = "1.2.2.0/24 network1 secret3 3",
   Framed-Route = "1.2.1.3 user3 secret OSPF 5",
   Framed-Route = "1.2.1.4 user4 secret OSPF 5",
   Framed-Route = "1.2.1.5 user5 more_secret RIP 5",
   Framed-Route = "1.2.1.6 user6 secret6 RIP6",
   Framed-Route = "1.2.1.7 user7 secret7 RIP 7",
   Framed-Route = "1.2.1.8 user8 secret8 RIP8",
   Framed-Route = "1.2.1.9 user9 secret9 RIP9",
   Framed-Route = "1.3.1.0/24 network10 passwdnetwork10 10",
   Framed-Route = "1.3.2.0/24 network11 passwdnetwork11 11",
   Framed-Route = "1.3.3.0/24 network12 passwdnetwork12 12",
   Framed-Route = "1.3.4.0/24 network13 passwdnetwork13 13",
   Framed-Route = "1.3.5.0/24 network14 passwdnetwork14 14",
   Framed-Route = "1.4.6.0/24 network15 passwdnetwork15 OSPF 15",
   Framed-Route = "1.4.2.0/24 network16 passwdnetwork16 OSPF 16",
   Framed-Route = "1.4.2.0/24 network17 passwdnetwork17 OSPF 17",
   Framed-Route = "1.4.2.0/24 network18 passwdnetwork18 18",
   Framed-Route = "1.4.2.0/24 network19 passwdnetwork19 RIP 19",
   Framed-Route = "1.5.1.0/24 network20 passwdnetwork20 RIP 20",

dialout-2
   ..............................
   ..............................
dialout-3
   ..............................
   ..............................
```

Specifying the BRICK to which the IP routing information should go

In the event that you have more than just one BRICK to which your IP routing information is to be transferred, it is possible to differentiate between the routers using the following syntax for the aforementioned dummy user, the following **sysName** variable is from the MIB II table **system**:

```
dialout-[sysName]-x
dialout-brick1-1
   ..............................
   ..............................
dialout-brick1-2
   ..............................
   ..............................
dialout-brick1-3
   ..............................
   ..............................
dialout-brick2-1
   ..............................
   ..............................
dialout-brick2-1
   ..............................
   ..............................
dialout-brick2-3
   ..............................
   ..............................
```

When the IP routing information is loaded to the BRICK (usually on booting when **RadiusSrvDialout** is set to *enabled*), the information is stored in the **ipRouteTable.** The indices for the as yet unused interfaces for these route entries extend from 30000. The **ipRouteTable** could look something like this:

```
inxDest(*rw)Ifindex(rw)Metric1(rw)Metric2(rw)
   Metric3(rw)Metric4(rw)NextHop(rw)Type(-rw)
   Proto(ro)Age(rw)Mask(rw)Metric5(rw)
   Info

03 1.2.1.1  30001 3        0
   0        1     0.0.0.0  indirect
   other    1538  255.255.255.250
   .0.0
04 1.2.1.2  30002 3        0
   0        3     0.0.0.0  indirect
   other    1540  255.255.255.2550
   .0.0
05 1.2.2.0  30003 3        0
   0        3     0.0.0.0  indirect
   other    1540  255.255.255.2550
   .0.0
```

Example: Propagating dial-out IP routes via RIP

Here an example for propagating dial-out IP routes via RIP:

```
dialout-1
   Framed-Route = destaddr/mask userid userpw RIP
```

For settings made in the **ipExtIfTable** on the BRICK and derived from the BinTec-specific Radius attributes, the variable **RouteAnnounce** is rendered ineffectual.

In principle, this is possible but not advisable if there is a large number of IP routes.

Example: Propagating dial-out IP routes via OSPF

Here an example for propagating dial-out IP routes via RIP:

```
dialout-1
   Framed-Route = destaddr/mask userid userpw OSPF
```

For settings made in the **ipExtIfTable** on the BRICK and derived from the BinTec-specific Radius attributes, the variables **Ospf**, **RouteAnnounce** and **OspfMetric** are rendered ineffectual. The dial-out IP routes are thus propagated with an OSPF metric calculated as follows:

IpMetric + 20

Example: Dial-out IP routes and Proxy-ARP

Here an example for dial-out IP routes and Proxy-Arp:

```
dialout-1
```

```
Framed-Route = destaddr/mask userid userpw PROXYARP
```

For settings made in the **ipExtIfTable** on the BRICK and derived from the BinTec-specific Radius attributes, the variable **ProxyArp** is rendered ineffectual.

Partner-specific information

Now it is necessary to assign specific details about the partner to each IP route entry, again in the users file of the RADIUS server. Here it is possible that several routes refer to just the one user entry. The minimum configuration entries must include the following:

Attribute	Meaning
Service-Type = Framed	This is the default value for PPP connections.
Framed-Protocol = PPP	This is the type of encapsulation used. If not set PPP is used.
Framed-IP-Address = X.X.X.X	This is the IP address of the WAN partner and must correspond to the destination address in the routing entry described above in Framed-Route.
Framed-IP-Netmask = Y.Y.Y.Y	IP netmask, it could be something like *255.255.255.255.*
BinTec-biboDialTable = "direction=outgoing number=********"	A temporary entry for dial-out, including the phone number of the WAN partner, is made in the **biboDialTable**.
Password	A user password that must match the password used in the Framed-Route (see **userpw** above).

Table A-15: Attributes for partner-specific information (minimum configuration entries)

Caution!

It is important for security reasons to make sure that no incoming calls are authenticated over this entry.

➤ Set **direction** to *outgoing.*

The following attributes are optional:

Attribute	Meaning
Framed-MTU	Sets the **ifMtu** variable in the **IfTable**.
Framed-Compression	Sets the **VJHeaderComp** variable of the **PPPTable** if necessary.
Idle-Timeout	Sets the **ShortHold** variable in the **PPPTable**.
Port-Limit	Sets the **MaxConn** variable in the **PPPTable**.
BinTec-biboPPPTable = "biboPPPAuthentica-tion=pap/chap/ms_chap"	The authentication protocol used for dial-out; if this is not explicitly specified, the authentication protocol CHAP is set.
BinTec-biboPPPTable = "biboPPPLocal-Ident=local_pppid"	This is the local ppp ID for authentication at the WAN partner´s (optional) and the default set-ting.

Table A-16: Optional attributes for partner-specific information

Example For an example of Framed-Route and corresponding partner-specific entries in the users file see section A, chapter 1.7.7, page 46.

1.7 Examples

1.7.1 Typical Dial-In (Without BinTec Attributes)

To enter a user for typical dial-in, there has to be an entry like the following in the RADIUS database (users file):

```
userPassword = topsecret,
   Service-Type = Framed,
   Framed-Protocol = PPP,
   Framed-IP-Address = 1.2.1.1,
   Framed-IP-Netmask = 255.255.255.255,
   Idle-Timeout = 25
```

1.7.2 Standard Dial-In with CLID

To identify RADIUS partners outband by their CLID (calling line identification, i.e. ISDN telephone number) there has to be an entry like the following in the RADIUS database (users file):

```
userPassword = topsecret3,
   Service-Type = Framed,
   Framed-Protocol = PPP,
   Framed-IP-Address = 1.2.1.1,
   Framed-IP-Netmask = 255.255.255.255,
   Idle-Timeout = 25,
      BinTec-biboDialTable = "number=00815123456
                              direction=outgoing"
```

Note that the phone number must be specified here exactly as it is signalled with the incoming call (you can see this in the **RemoteNumber** field of the **isdnCallTable**).

When a call from the number 00815123456 comes in, a new PPP entry is generated with **Encapsulation** = *PPP*.

Please also note that when using RADIUS inband authentication it can take up to 2 seconds to accept an incoming call if the RADIUS server is delayed inactive.

1.7.3 Callback PPP Negotiated

To configure callback PPP negotiated, there has to be an entry like the following in the RADIUS database (users file):

```
userPassword = topsecret,
   Service-Type = Callback-Framed,
   Framed-Protocol = PPP,
   Framed-IP-Address = 1.2.1.1,
   Framed-IP-Netmask = 255.255.255.255,
   Idle-Timeout = 25
      Callback-Number = 12345
```

1.7.4 Callback (Windows Client)

There are several possibilities:

- Callback to the Windows client will take place in every case, the client has to enter the number to be called during the negotiation.
  There has to be an entry like the following in the RADIUS database (users file):

```
msclient password = xy,
         Service-Type = Callback-Framed,
         Framed-Protocol = PPP,
         Idle-Timeout = 600,
         biboPPPTable = "MaxRetries=1"
```

You can handle the whole configuration even without using BinTec specific attributes at all. But we suggest using the entry `biboPPPTable = "MaxRetries=1"` for the case that the Windows user enters a wrong phone number for callback.

- Callback to the Windows client will take place in every case, the number is entered on the BinTec router.
  There are two possibilities:
  There has to be an entry like the following in the RADIUS database (users file):

```
msclient password = xy,
         Service-Type = Callback-Framed,
         Framed-Protocol = PPP,
         Idle-Timeout = 600,
         Callback-Number = 12345
```

  Or with BinTec attributes:

```
msclient password = xy,
         Framed-Protocol = PPP,
         Idle-Timeout = 600,
         biboPPPTable = "callback=ppp_offeredMaxRetries=1",
         biboPPPTable = "Authentication=ms_chap
                        AuthSecret=xx",
         biboDialTable ="number=12345 direction=outgoing"
```

- Callback to the Windows client is allowed but the user has the possibility to enter a number or to cancel the callback.
  There has to be an entry like the following in the RADIUS database (users file):

```
msclient password = xy
         Framed-Protocol = PPP,
         Idle-Timeout = 600,
         biboPPPTable = "callback=callback_optional
                        MaxRetries=1",
         biboPPPTable = "Authentication=ms_chap
                        AuthSecret=geheim"
```

1.7.5 Callback (CLID)

RADIUS server

To configure Callback (CLID), there has to be an entry like the following in the RADIUS database (users file):

```
9119732123Service-Type = Framed,
           Framed-Protocol = PPP,
      Framed-IP-Address = 1.2.1.1,
      Framed-IP-Netmask = 255.255.255.255,
      Idle-Timeout = 25
            Reply-Message = username
```

BinTec router

In addition on the BinTec router, the variable **biboPPPProfileAuthRadius** in the **biboPPPProfileTable** has to be set to *outband* (see "biboPPPProfile-Table", page 17).

1.7.6 Working with one or more RADIUS Servers

In this example, you can see how to work with more than one RADIUS server.

Here are examples of two different entries in the **RadiusServerTable**:

```
inx Protocol(*rw)    Address(rw)    Port(rw)       Secret(rw)
    Priority(rw)     Timeout(rw)    Retries(rw)    State(-rw)
    Policy(rw)       Validate(rw)   Dialout(rw)    DefaultPW(rw)

00  authentication   172.16.70.14   1645           secret
    0                1000           5              active
    authoritative    enabled        enabled

01  authentication   172.16.70.93   1645           secret
    1                1000           5              active
    authoritative    enabled        enabled
```

What happens on the BRICK? According to the example above, once a dial-out request is made that is to be sent on one of the routes loaded from the Radius server and thus occupying an **IfIndex** above 30000, the Radius server with the IP address 172.16.70.14 receives a request, as this entry has the lowest **RadiusServerPriority** setting.

Backup If this server does not reply after 5 attempts (**RadiusSrvRetries**), each after an interval (**RadiusSrvTimeout**) of 1000 seconds, the **RadiusSrvState** is set to *inactive*. The server with the next lowest priority setting, in this case the server with the IP address *172.16.70.93*, then receives a request from the BRICK. If this server responds to the BRICK request, the partner-specific information for an outgoing call to a WAN partner can be loaded from the Radius server to the corresponding MIB tables on the BRICK.

1.7.7 Dial-Out

The following example shows a Framed-Route with the corresponding partner-specific entries in the users file:

```
dialout-1
   Framed-Route = "1.2.1.1 user1 topsecret3"

user1Password = topsecret3,
   Service-Type = Framed,
   Framed-Protocol = PPP,
   Framed-IP-Address = 1.2.1.1,
   Framed-IP-Netmask = 255.255.255.255,
   Idle-Timeout = 25,
   BinTec-biboPPPTable = "biboPPPAuthentication=chap",
   BinTec-biboPPPTable = "biboPPPLocalIdent=mylocalid",
   BinTec-biboDialTable = "direction=outgoing
                              number=00815123456"
```

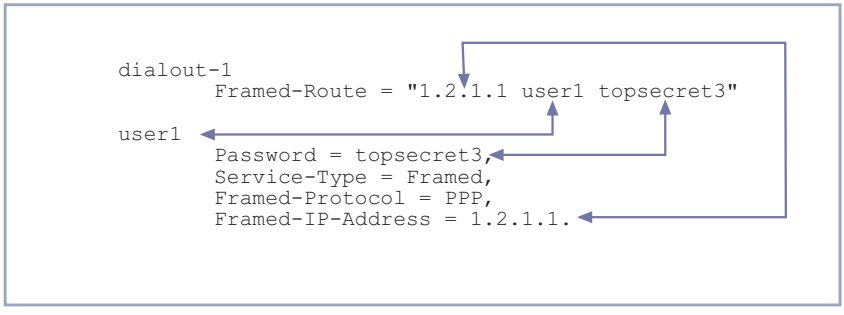

Figure A-1: Matching Framed-Routes and partner-specific entries

For the purpose of clarification, this example places the Framed-Route together with the partner-specific information. As shown above, user name, IP address and password are identical in the routing and partner-specific information included in each example. This is essential for RADIUS for dialout to function properly.