

RELEASE NOTE BIANCA/BRICK-XS/ XS OFFICE

July 24, 1998

New System Software:

Release 4.8 Revision 6

This document describes the new features, enhancements, bugfixes, and changes to the BIANCA/BRICK-XS/ XS office System Software since Release 4.8 Revision 3 (current user documentation).

Upgrading System Software	2	
What's New in Release 4.8.6	3	
Known Problems	3	
Features	3	
Microsoft Callback Extension to Mode 3	3	
Microsoft Callback via RADIUS	4	
IPX RADIUS Extensions	4	
X.25	4	
Performance Enhancement	9	
IP Filter for TCP State and ICMP Type	9	
New Trace Command Feature	11	
Status Display for Modems	11	
Wildcards for Dialing Numbers	12	
Link Quality Monitoring	13	
Extended Syslog Messages	13	
Bugfixes	13	
Access Lists	13	
TPO Bridge	14	
X.31 in D-Channel	14	
CAPi	14	
Dynamic Shorthold	15	
STAC Compression on Multilink PPP Interfaces	15	
Spaces in biboPPPLoInString	15	
Setting Administration Status to Down	16	
Setup Tool	16	
Detailed Feature Descriptions	17	
IPX RADIUS Extensions	17	
Link Quality Monitoring	20	
What Was New in Release 4.8.3	22	

Upgrading System Software

1. Retrieve the current system software image from BinTec's WWW server at <http://www.bintec.de>.
2. With this image you can upgrade the BIANCA/BRICK-XS/ XS office with the **update** command from the SNMP shell via a remote host (i.e. using telnet, minipad, or isdnlogin) or by using the **BOOTmonitor** if you are logged in directly on the console.

Information on using the BOOTmonitor can be found in the *BRICK-XS/ XS office User's Guide* under *Firmware Upgrades*.

3. Once you've installed Release 4.8 Revision 6 you may want to retrieve the latest documentation (in Adobe's PDF format), which is also available from BinTec's FTP server at the address noted above.

Note: When upgrading system software, it is also recommended that you use the most current versions of *BRICKware for Windows* and *UNIXTools*. Both can be retrieved from BinTec's FTP server.

Info:



Performing a software update on a running system (via the **update** command) currently requires that a contiguous block of free memory, \geq the size of the new software image, is available.

To verify enough memory space is available use the **show mem** command and note the output of the "largest block" field. To maximize free memory two options are available.

- Perform the update immediately after rebooting the system. This ensures that memory has been defragmented.
- Temporarily reduce the size of your configuration file by deactivating memory intensive software options such as OSPF or IPX.

Note that you can always perform an update using the BOOTmonitor. The internal procedure of performing software upgrades on the BRICK is currently being optimized and a change is planned for a future release.

What's New in Release 4.8.6

Release 4.8 Revision 6:

Released: 24.07.98

Features:

Bugfixes:

Detailed Description:

Known Problems

In Release 4.8 Revision 6, two problems currently exist.



1. Network Address Translation

After receiving several broadcast packets via an interface where NAT is being performed the BRICK may either “lock-up” or inadvertently reboot. If the system locks up the BRICK will no longer be accessible (via remote or console) and must be power cycled on and off.

2. Dial-up connections for RADIUS-Users

Interfaces configured to use `ip_lapb` encapsulation, using the following entry in `/etc/raddb/users`,

```
BinTec-biboPPPTable = "Encapsulation=ip_lapb"
are sometimes rejected by the BRICK.
```

Features

Microsoft Callback Extension to Mode 3

The Microsoft Callback Control Protocol (CBCP) knows different modes to decide which number is used for callback. This protocol is activated, when there is a call from a Windows95/NT client.

Up to now Mode 2 was implemented. In Mode 2 (callback to a user-defined number) the user is asked, when calling from a Windows95/ NT client, to enter the callback number. This number is then used for callback.

From this release on the MS-CBCP was extended to Mode 3. Mode 3 uses a predefined number for callback.

Which mode is used (Mode 2 or Mode 3) depends on whether there is a predefined number assigned. When there is a predefined number, either a entry in the `biboPPPDialTable` for this partner (*Direction: both or outgoing; Type: isdn or isdn_spv*) or

when authentication is made via RADIUS and the RADIUS attribute *Callback-Number* is assigned, then Mode 3 is used. When calling from a Windows95/ NT client the caller is asked in a dialog box to confirm the mode (Mode 3) respectively the callback number. With no number assigned callback is made using Mode 2.

Such it is ensured that a callback is either made using the user-specified or the predefined number.

The variable *CallBack* in the *biboPPPTable* can be set to *ppp_offered* or *enabled*. But you must notice that with the value set to *enabled* no authentication is made during callback.

Also see Microsoft Callback via RADIUS below.

Microsoft Callback via RADIUS

With this new release it is possible to use Microsoft Callback via RADIUS for calls from a Windows95/ NT client.

The RADIUS server must be configured as follows:

Service-Type = Callback-Framed

Specifying only the Service-Type means using Mode 2 of the CBCP (user-specifiable number). This configuration assigns the value *enabled* to the variable *biboPPPCallBack* in the *biboPPPTable*.

To use Mode 3 (predefined number), that means using a fixed callback number, you must additionally assign a callback number as in the following example:

Service-Type = Callback-Framed

Callback-Number = "392"

The feature Microsoft Callback via RADIUS is only available for an inband identification of the caller (no calling line identification). The same it's not possible to set the value *ppp_offered* for the variable *biboPPPCallBack*. For the time of the PPP connection there exists a temporary entry in the *biboPPPTable* with variable *CallBack* assigned the value *enabled*. This means that there is no additional authentication during callback. In

Mode 3 a temporary entry in the *biboPPPDialTable* using the defined calling number is generated, too.

IPX RADIUS Extensions

The BRICK now supports dial-up IPX client connections via RADIUS. For a detailed description of this new feature see IPX RADIUS Extensions on page 17.

X.25

X.25 Window/ Packet Size Negotiation

Now you can decide for each X.25 link, whether a window/ packet size negotiation is made.

x25LkPrNegotiation is the new parameter in the *x25LinkPresetTable*, which handles this feature. This parameter can be assigned three possible values:

<i>never</i>	No negotiation. When a call arrives that does not correspond to the default size, the call is cleared.
<i>always</i>	Negotiations are always made.
<i>when_necessary</i>	There are only negotiations, when the requested values differ from the default values.

Window/ packet size negotiation settings can also be configured via Setup Tool, see “[Configuring X.25 Parameters in Setup Tool](#)” on page 5.

Configuring X.25 Parameters in Setup Tool

Now it is possible to configure additional X.25 parameters using Setup Tool.

- X.25 Link Configuration



Here window/ packet size negotiation can be adjusted for an X.25 link. **Window size/ Packet size Neg.** corresponds to the parameter *x25LkPrNegotiation* in the *x25LinkPresetTable*.

BIANCA/BRICK-XS Setup Tool		BinTec Communications GmbH	
[X.25][LINK][ADD]: X.25 Link Configuration		mybrick	
Link	en1-llc		
L3 Mode	dte		
L3 Window Size	default: 128	max: 128	
L3 Packet Size	default: 2	max: 7	
Window size/ Packet size Neg.	when necessary		
Lowest Two-Way-Channel (LTC)	1		
Highest Two-Way-Channel (HTC)	2		
Partner MAC Address (LLC)			
Layer 2 Behaviour	disconnect when idle		
SAVE		CANCEL	
Use <Space> to select			

Window size/ Packet size Neg. = Decides whether window/ packet size negotiation is made for this X.25 link. The possible values are **never**, **always** and **when necessary**, where **when necessary** is the default value. The value *never* means no negotiation. When a call arrives that does not correspond to the default size, the call is cleared. *Always* means negotiations are always made and when *when necessary* is selected, there are only negotiations, when the requested values differ from the default values.

•WAN Partner



For WAN partners using the protocols X.25, X25ppp, X.31 B-Channel or X.25 no signalling the Layer 2 Mode can be configured in the advanced settings. The item Layer 2 Mode corresponds to the parameter *biboPPPLayer2Mode* in the *biboPPPTable*.

BIANCA/BRICK-XS Setup Tool		BinTec Communications GmbH	
[WAN][EDIT][ADVANCED]: Advanced Settings		mybrick	
Callback	no	Static Shorthold	20 Idle for Dynamic Shorthold (%)0
Delay after Connection Failure	300	Dynamic Name Server Negotiation	yes
Channel-Bundling	no		
Layer 1 Protocol	ISDN 64 kbps	Layer 2 Mode	dte
OK		CANCEL	
Use <Space> to select			

Layer 2 Mode = Layer 2 Mode can receive the values **auto**, **dte** or **dce**, where **auto** is the default value

•WAN Partner Numbers Advanced



Here the item Closed User Group can be configured. The item corresponds to the parameter *biboDialClosedUserGroup* in the *biboDialTable*.

BIANCA/BRICK-XS Setup Tool		BinTec Communications GmbH	
[WAN][Extended]: Extended Settings of WAN-Partner Numbers		mybrick	
Closed User Group		none	
OK		CANCEL	
Use <Space> to select			

Closed User Group = The item Closed User Group can be assigned the values **none** or an integer from **1 to 9999**. **None** is the default value.

Active Layer 2 Set Up for Incoming X.25 Calls

Prior to release 4.8.6 the BRICK remained passive during setup of incoming X.25 dialup connections and waited for a SABM (Set Asynchronous Balance Mode) from the caller. Some X.25 implementations however were also waiting for a SABM from the BRICK.

Now the BRICK only waits one second for an incoming SABM. If no SABM is received within this time, the BRICK will send a SABM.

Because of the wait time the probability of a layer 2 setup collision is very small. Standard end-devices handle such collision correctly.

TP0 Bridge Extensions

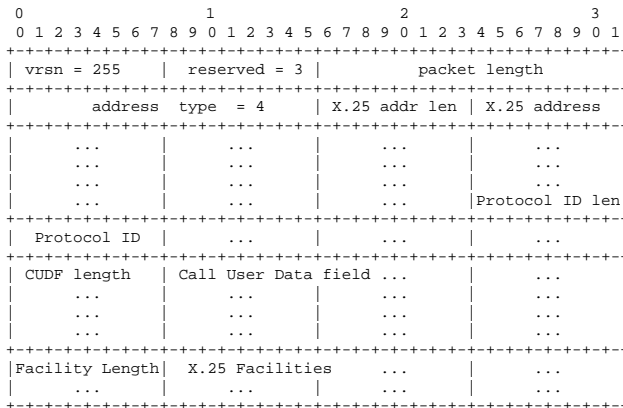
To make possible connections between TCP clients and an X.25 network the TP0 Bridge feature (RFC 1086/ RFC 1006) is implemented on the BRICK.

With this release two extensions concerning the transmission of X.25 data (RFC 1006) for incoming X.25 connections to the TP0 Bridge have been made.

Firstly with release 4.8.6 NSAP addresses, which are subaddresses of X.25 addresses, can be proofed for incoming X.25 calls. If a listener transfers a NSAP address in the facility field of the listening address, only X.25 calls with the same NSAP address are signaled to the listener.

The second extension concerns the X.25 call indication packet, which is sent as the first packet, when an incoming X.25 connection is established. Now with release 4.8.6 there is a possibility that the listening application gets some information about the contents of the X.25 call indication packet.

To get this data the value of the function byte, (the first byte, the listener sends to the TP0 bridge, see RFC 1086) has to be 66 instead of 2. Then the first data packet, the listener receives on its new established TCP stream has the following format: It consists of 4 Byte TP0 header and the data in the extended X.25 Address Format:



Performance Enhancement

Release 4.8 Revision 6 contains additional internal performance enhancements that greatly reduce system load on systems supporting V.110 and Modem connections over PPP. With these enhancements the system load on the BRICK has been reduced by 50% (compared to previous releases with the same throughput).

IP Filter for TCP State and ICMP Type

The filters for IP access have been enhanced.

ICMP Type

The filters can now be used to filter IP packets in dependence of the ICMP type.

In the *ipFiltertable* there is the new variable *icmptype*, which can be assigned the following values :

echoRep, *destUnreach*, *srcQuench*, *redirect*, *echo*, *timeExcds*, *parmProb*, *timestamp*, *timestampRep*, *addrMask*, *addrMaskRep*, *dont_verify* .

Setup Tool's Filters menu has also been changed. You can now define filters according to appropriate ICMP types using the Type field after setting the protocol field to "ICMP".



BIANCA/BRICK-XS Setup Tool		BinTec Communications GmbH	
[[IP]][ACCESS][FILTER][ADD]: Configure IP Access Filter		mybrick	
Description	echo request		
Index	9		
Protocol	icmp		
Type	echo		
Source Address			
Source Mask			
Source Port	any		
Destination Address			
Destination Mask			
Destination Port	any		
SAVE		CANCEL	
Use <Space> to select			

TCP Connection State

Filters can now be defined based on the state of an TCP Connection.

In the *ipFilterTable* there is the new variable *TcpConnState*, which can be assigned the following values:

dont_verify, *established*.

When this variable is set to *established*, this filter matches for TCP packets, which do not initiate a connection.

A typical application for this filter is to let packets pass through, which belong to connections that were initiated from inside, but discard all other TCP packets. This can be configured by the following rules:

1. rule: ALLOW (TCP/ established)
2. rule: DENY (TCP/ dont_verify)

The configuration in Setup Tool:



BIANCA/BRICK-XS Setup Tool		BinTec Communications GmbH	
[IP][ACCESS][FILTER][ADD]: Configure IP Access Filter		mybrick	
Description	TCP established		
Index	10		
Protocol	tcp		
Connection State	established		
Source Address			
Source Mask			
Source Port	any		
Destination Address			
Destination Mask			
Destination Port	any		
SAVE		CANCEL	
Use <Space> to select			

New Trace Command Feature

The trace command has been enhanced. It is now possible to decode HOLD and RETRIEVE messages in the D-Channel.

Status Display for Modems

This feature concerns the BIANCA/BRICK-XS office only.

The web based status page for your BRICK now additionally displays information on installed modems.

Hardware Interfaces

LAN	Ethernet	o.k.	
WAN	ISDN SO	o.k.	ISDN: used 1, available 1 ● ● Modem 14.4: used 1, available 0 ●
LOCAL			connected FX 14400/TX 14400 ISDN 3001

Under Hardware Interfaces you will find the modems with the respective slot they are installed in. The modem type and

modem status are displayed. Each modem used at the moment is marked red and when you move the mouse pointer over the red channel symbol, the rate for receiving and transmitting data in bps is displayed. Additionally you are informed about the ISDN channel used. The four digits *xyzz* stand for the slot (*x*), the unit (*y*) and the ISDN channel used (*zz*). For the example below this would consequently mean: slot 2, unit 0 and channel 1.

Wildcards for Dialing Numbers

Similar to wildcards for the calling party's address in the *biboDialTable* for incoming calls, the variable *Number* now can also contain wildcards for outgoing calls. The wildcards for outgoing and incoming calls are defined as follows:

Wildcard	Example	Outgoing Calls	Incoming Calls
*	1234*	is ignored, e.g. 1234	matches zero or any string, e.g. 1234 or 123467
?	1234?	is replaced by 0, e.g. 12340	matches any single digit, e.g. 12349, 12347
[a-b]	123[5-9]	first digit in the range, e.g. 1235	denotes the range of possible digits to match, e.g. 1235, 1236
[^a-b]	123[^0-5]	range of digits not allowed, first possible digit inserted, e.g. 1236	denotes the range of excluded digits to match, e.g. 1236, 1237
{ab}	{00}1234	inserted for outgoing calls, e.g. 001234	optional string to match, e.g. 001234, 1234

The advantage is, that now you can use one entry for the variable *Number* for incoming and outgoing calls. For Example {0}91196790 will generate 091196790 for outgoing calls and will accept 091196790 and 91196790 for incoming calls as valid CLID.

Link Quality Monitoring

By the help of Link Quality Monitoring (LQM defined in RFC 1989) it is possible to exchange information within a PPP connection to draw conclusions about the underlying connection quality.

This information is typically transmitted periodically to the partner as so-called Link Quality Reports (LQR). The interval (Reporting Period) is agreed upon during the LCP negotiation.

Link Quality Monitoring can be useful to examine e.g. modem connections. (With unreliable modem connections it can happen that because of CRC errors no more data can be transmitted.)

For detailed information on the new feature Link Quality Monitoring see the section Detailed Feature Descriptions on page 20.

Extended Syslog Messages

This feature concerns the BIANCA/BRICK-XS office only.

The syslog messages have been extended for a detailed analysis of fax connections.

Bugfixes

Access Lists

- There was a bug in the access lists implementation concerning the following conditions:
 1. Filtering for source or destination ports.
 2. The action is deny
 3. The IP datagrams are fragmented.

Under these conditions fragments are discarded, though the ports of the complete datagram do not correspond to the ports defined in the filters.

This bug has been fixed.

TP0 Bridge

- Sometimes it happened that it was not possible to establish an initial TP0 Bridge connection (RFC 1086) between the TCP client and the BRICK.

This bug has been fixed. In this context the TP0 Bridge syslog messages have been changed, too.

X.31 in D-Channel

- When configuring TEI values for X.31 in D-Channel by the autoconfiguration for the BRICK's ISDN interface, it sometimes happened that wrong TEI values were generated in the *isdnDChanX31Table*.

This bug has been fixed.

- When CAPI applications were using X.31 in D-Channel for X.25 packet switching, the variable *AssignedTo* in the *isdnDChanX31Table* was not considered (Setup Tool: Advanced settings for the WAN interface).

This problem has been fixed.

The variable *AssignedTo* can be assigned the following values:

<i>packet_switch</i>	The TEI may be used only by the X.25 router.
<i>capi</i>	The TEI may only be used by a CAPI application and this TEI has to be configured also in the CAPI application.
<i>capi_default</i>	The TEI may be used only by a CAPI application and the BRICK overwrites the TEI value that is configured in the application.
<i>delete</i>	Sets the table entry to delete.

CAPI

- If a CAPI application used an X.25 protocol, it wasn't any longer informed about incoming X.25 calls after it has sent

a CONNECT_B3_REQ message to establish an outgoing X.25 connection. In this case the CONNECT_B3_Ind message, the application has to receive got lost.

This bug has been fixed.

Dynamic Shorthold

- When combining dynamic B-Channel Bundling with optimally making use of the charging intervals (by dynamic shorthold), there was the problem that, when reducing the bandwidth, the current charging intervals were not taken into consideration. This problem has been fixed.
To optimize charges now the bandwidth is reduced by disconnecting a B-Channel only short before a new charging interval.

STAC Compression on Multilink PPP Interfaces

- According to RFC 1974 (*PPP STAC LZS Compression Protocol*) there are several check modes to keep the compressor and decompressor histories in synchronisation even in the absence of a reliable link to guarantee the sequential transmission of data.

In Release 4.8.3 it still happened that in rare cases, when *bibopppCompression* = stac (RFC 1974, check mode 3) was used on MultiLink PPP interfaces the history re-synchronisation process sometimes came to a state where decompression histories were out-of-sync and user data could no longer be transmitted over the line.

This problem has been fixed in the current release.

Spaces in biboPPPLoginString

- There appeared problems in the login procedure configuration (especially for Compuserve users), when strings like passwords or login names were containing spaces. That was because spaces are used as internal flags to handle the login procedure.

To handle this problem blank spaces in strings, which are part of the variable *LoginString* in the *biboPPPTable*, must be preceded by a backslash as shown in the following example for the string “pass word”:

```
inx LoginString(rw)
```

```
00 "-d1 \n e: CIS\n D: name/go:pppconnect\n wor -d1 pass\ word\n PPP"
```

This must be considered, when the variable *LoginString* is configured via SNMP. In the Setup Tool no additional backslashes have to be entered, when configuring the items Host, User ID and Password in the menu [WAN][EDIT][ADVANCED][PROVIDER].

Setting Administration Status to Down

- When by “ifconfig down” or via the Setup Tool the variable *ifAdminStatus* was set to down for an active interface, the variables in the PPP accounting syslog message containing PPP connection information only had the value 0.

This problem has been solved. Now the syslog message contains the correct values.

Setup Tool

- When a PPP interface was resetted in the interface monitor in [MONITOR][INTERFACES][EXTENDED], there could occur a reboot of the BRICK with large configurations.

This problem has been solved.



- When a default route was configured for a WAN partner interface in the [IP][ROUTING] menu or the SNMP shell and afterwards the [WAN][EDIT][IP] menu was opened again for this WAN Partner and left with SAVE, this default route was deleted.

This bug was fixed.

Detailed Feature Descriptions

IPX RADIUS Extensions

The BRICK now supports dial-up IPX client connections via RADIUS. Support for IPX links via RADIUS has been tested using Merit's AAA RADIUS server 3.5.6. The examples shown below can be used with the Merit server, for use with other servers consult your local documentation.

Assuming a RADIUS server has been configured in Setup Tool's  →  menu (or the *radiusServerTable* from the SNMP shell), and the RADIUS server can successfully authenticate the caller, dial-up links can be setup to support IPX networking using standard IPX or BinTec-specific RADIUS attributes mentioned below.

Standard Attributes

Framed-IPX-Network

This attribute defines a transfer network for the IPX link. Setting this attribute to "Framed-IPX-Network = 8" effectively sets the IPX network number for the transfer network to "0:0:0:8".

Normally, when this attribute is set to "ffffffe" the calling host is assigned a network number from an existing address pool, currently this feature is not supported on the BRICK. To disable a transfer network for the IPX WAN link you can set this attribute to "0" (no transfer network, unnumbered RIP).

RIP/SAP updates: If the "Framed-IPX-Network" attribute is used entries in the BRICK's *ripCircTable* and *sapCircTable* are configured using default settings for RIP/SAP updates (triggered+piggybacked).

BinTec-specific Attributes

BinTec-specific RADIUS attributes added in release 4.8.3 are defined below. For additional information, see also the section [Using BinTec-specific Attributes](#) on page 19.

BinTec-ipxCircTable Creates or modifies *ipxCircTable* entries.

BinTec-ripCircTable Creates or modifies *ripCircTable* entries
 BinTec-sapCircTable Creates ore modifies *sapCircTable* entries.

Example IPX over RADIUS /etc/raddb/client Entries

The definitions below could be used for a dial-up IPX link between two BRICKs. RIP/SAP updates will be performed via the default triggered+piggybacked mode.

```
kornburgxl
  Password = "access2roth", Framed-IPX-Network = 0
  Framed-Protocol = PPP, Idle-Timeout = 300,

rothxl
  Password = "access2kornburg", Framed-IPX-Network = 0
  Framed-Protocol = PPP, Idle-Timeout = 300,
```

If the router rothxl supports IPX WAN links but requires a transfer network (0:0:0:9) this definition could be used instead.

```
rothxl
  Password = "access2kornburg", Framed-IPX-Network = 9
  Framed-Protocol = PPP, Idle-Timeout = 300,
```

If the required IPX services are statically configured, RIP and SAP can be disabled for the WAN link using the BinTec-specific options shown below.

```
kornburgxl
  Password = "access2roth", Framed-IPX-Network = 0
  Framed-Protocol = PPP, Idle-Timeout = 300,
  BinTec-ripCircTable = "ripcircstate=off",
  BinTec-sapCircTable = "sapcircstate=off"
```

The definition below could be used for a dial-in Windows Client that does not support IPX WAN links. In this example, setting "Update = 0" disables periodic updates for active links.

```
winlaptop
  Password = "486pentium?"
  Framed-Protocol = PPP, Idle-Timeout = 300,
  Bintec-ipxCircTable = "netnumber=0:0:0:a ipxcirctype=ipxcpWS"
  BinTec-ripCircTable = "Update=0 AgeMultiplier=10000",
  BinTec-sapCircTable = "Update=0 AgeMultiplier=10000"
```



RIP/SAP Updates for RADIUS interfaces

Since RADIUS interfaces are only available as long as the re-

spective client is connected please note the following effects this may have on links configured for active RIP and SAP updates.

- Access to services on the dial-in client's LAN (from hosts on the RADIUS client's LAN) may not be reliable.
- IPX clients cannot be informed of changes (routing or service advertisements) unless they are actually connected when the change occurs.

This may lead to a state where a server appears as being present on the remote network but is no longer available. The preferred solution, albeit time-consuming, to this problem is to statically configure the required routes and services on the clients and to disable RIP/SAP updates.

Using BinTec-specific Attributes

Each of the BinTec-specific RADIUS attributes corresponds to a MIB table. Supported BinTec-specific attributes can be used in your server's /etc/raddb/users file. The attribute definitions must also be added to your dictionary file (normally found in /etc/raddb). To modify a MIB table entry you must use the following syntax:

<BinTec-Option> = "variable1=value1 ... variablen=valuen"

An example authentication line from a RADIUS /etc/raddb/users file might look like this:

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

Also, when using these attributes please note:

- The table entry's *ifIndex* is set automatically and can't be influenced.
- The entries are not case-sensitive.
- You must not use blank spaces before or after »« signs inside the double quotes.
- Attributes support either **static** or **dynamic** mode.

Static mode modifies existing table entries while dynamic mode creates a new table entry. All variables you want to create (dynamic) must be defined in one line.

Link Quality Monitoring

Because Link Quality Monitoring as described under Features is specified within LCP negotiation, i.e. before the authentication of the partner, for the configuration of incoming calls a distinction must be made between inband and outband identification.

In case of outband identification (CLID/ outband RADIUS) and for outgoing calls the LQM is activated by setting the variable *biboPPPLQMonitoring* in the *biboPPPTable* to on. When a RADIUS server is used the variable is set by the help of the BinTec dictionary.

For incoming calls identified inband (identification by the internal *biboPPPTable* or via RADIUS server) the variable *biboPPPPProfileLQMonitoring* in the *biboPPPPProfileTable* must be set to on.

After a successful LCP negotiation for every link of a temporary connection additionally to the entry in the *biboPPPLinkTable* a correlating entry in the *biboPPPLQMTable* is generated. Both entries can be uniquely assigned to each other by the *IFIndex* respectively the *CallReference* value.

The *biboPPPLQMTable* is a new table and is described in detail in the following.

biboPPPLQMTable:

inx	lIndex(*ro	CallReference(ro)	ReportingPeriod(ro)
	OutLQRs(ro)	OutPackets(ro)	OutOctets(ro)
	InLQRs(ro)	InPackets(ro)	InOctets(ro)
	InDiscards(ro)	InErrors(ro)	PeerOutLQRs(ro)
	PeerOutPackets(ro)	PeerOutOctets(ro)	PeerInLQRs(ro)
	PeerInPackets(ro)	PeerInOctets(ro)	PeerInDiscards(ro)
	PeerInErrors(ro)	LossedOutLQRs(ro)	LossedOutPackets(ro)
	LossedOutOctets(ro)	LossedPeerOutLQRs(ro)	LossedPeerOutPkts(ro)
	LossedPeerOutOcts(ro)		

The *biboPPPLQMTable* contains statistical information for each current PPP link on the system. Only the system can add or delete entries to this table.

Entries are created by the system each time a new PPP link was established and LQM was negotiated successfully.

Entries are removed by the system, when the corresponding PPP link is disconnected.

For detailed information on the meaning of the single variables see the [MIB Reference](#) on the BinTec Website at <http://www.bintec.de>

What Was New in Release 4.8.3

Release 4.8 Revision 3:

Released: 24.04.98

Features:

Bugfixes:

Detailed Description:

Features

Monitoring Modem Connections

A new *mdmTable* has been added to the SNMP shell's Modem Group. The *mdmTable* contains one row entry for each modem detected on the BRICK . Each entry includes information regarding the current state of a specific modem. Status information details the detected modem type, whether a connection is established, and if so, the associated ISDN B-channel, the transmit/receive rates, as well as the negotiated compression and modulation modes.

Setup Tool has also been enhanced in Revision 4.8.3 and displays information retrieved from the new *mdmTable* in the **MONITORING AND DEBUGGING** → **MODEM** → menu as shown below.

Index	Action	Type	State	Mode	Modulation	ErrCompr	TX Corr	RX Speed	ifindex/	BChan
2000	enabled	mdm144	connected	ppp	v32bis	none	none	14K	14K	2000/1
2001	enabled	mdm144	idle	none	unknown	none	none	0	0	0/0
EXIT										

Press <Ctrl-n>, <Ctrl-p> to scroll

For detailed information regarding the new *mdmTable* or Setup Tool's [Monitoring and Debugging][Modem] menu please refer to the [BIANCA/BRICK MIB Reference](#) and Chapter 4 of the [BRICK-XS User's Guide](#) respectively (both are available via BinTec's web site).

Changes

PPP

- **Reporting of Dynamically Assigned IP Addresses:**

In previous releases syslog messages were generated when a remote client was assigned an IP address from the BRICK's local IP address pool (*biboPPPIpAssignTable*).

If a host-route is configured on the BRICK for the calling client, the BRICK always retrieves the address from the host route before checking the address pool. If an address was assigned in this manner a syslog message was not generated. Beginning in release 4.8.3 syslog messages are generated for all IP address assignments regardless of the method used.

Bugfixes

IP

- In previous releases telnet sessions from the BRICK to hosts supporting the "Telnet Data Encryption Option" (typically supported on BSD UNIX systems) was not possible. This problem has been fixed.
- In rare cases it wasn't possible to delete entries from the BRICK's *ipNatOutTable*. This problem has been fixed.

IPX

- IPX WAN links that were configured using the setting "piggyback (only if link active)" in the **Send RIP/SAP Updates** field in Setup Tool's [WAN Partners][EDIT][IPX] menu sometimes unexpectedly cause a system reboot when many changes occurred on the network and the

WAN link had not been active for a long time. This problem has been fixed.

PPP

- **PPP Accounting:**

In previous releases setting the *IfAdminStatus* object to “down” for an interface that was in the up state sometimes resulted in syslog accounting messages containing incorrect data. This problem has been fixed.

Before:

```
“dialup1: outgoing link closed, duration 0 sec,
  0 bytes received, 0 bytes sent, 0 charging units”
```

After:

```
“dialup1: outgoing link closed, duration 45 sec,
  2365 bytes received, 4347 bytes sent, 3 charging units”
```

- **STAC Compression on MultiLink PPP Interfaces:**

According to RFC 1974 (*PPP STAC LZS Compression Protocol*) there are several check modes to keep the compressor and decompressor histories in synchronisation even in the absence of a reliable link to guarantee the sequential transmission of data.

In rare cases, when *biboPPPCompression* = stac (RFC 1974, check mode 3) was used on MultiLink PPP interfaces the history re-synchronisation process sometimes came to a state where decompression histories were out-of-sync and user data could no longer be transmitted over the line.

Although the frequency of this problem has been greatly reduced in release 4.8.3 we currently recommend using MS-STAC compression (*biboPPPCompression=ms_stac*) when the remote partner supports this method. If the partner interface does not support MS-STAC, please note that this problem only occurs when MultiLink PPP is configured; and even then only in rare cases.

SNMP

A problem involving the *biboAdmLoginTable* has been fixed in revision 3. If a failed login attempt occurred on the BRICK for a user that was defined in this table (either an incorrect password or no password at all was entered) the BRICK automatically prompted with a new **login:** string. If the admin, read, or write user was then entered followed by the appropriate password, the BRICK incorrectly started the command (*biboAdmLoginCommand*) configured for the user from the failed login instead of the SNMP shell session. This problem has been fixed.

Detailed Feature Descriptions

For information regarding the new features included in System Software Release 4.8 Revision 3 please refer to the most recent versions of user documentation for the BIANCA/BRICK-XS.

User documentation is available (in Adobe's PDF format) via BinTec's Internet web site at:

http://www.bintec.de/ftp/brick_xs.html.