API Manual

User Database (Authentication Manager)

DOC220202APIV1.3.0.0EN | Revision V1.3.0.0 | English | Released | Public | 2023-07-07

empowering communication

Table of Contents

Chapter 1 Introduction

1.1 System Requirements

This software package has the following system requirements to its environment:

- netX90 (Use case C) Chip as CPU hardware platform

1.2 Intended audience

This manual is suitable for software developers with the following background:

- Knowledge of the netX DPM Interface ([1])

1.3 Terms, Abbreviations and Definitions

Term	Description
DPM	Dual Port Memory
LFW	Loadable Firmware
Password Hash	Cryptographic mechanism to transform a password text into an encrypted text to store in vulnerable system (e.g. filesystem)
User Management	Set of functions and modules that handle the User Database (e.g. add/delete users, factory reset)

Table 1. Terms, Abbreviations and Definitions

1.4 References to documents

This document refers to the following documents:

[1]	Hilscher Gesellschaft für Systemautomation mbH: Dual-Port Memory Manual, netX Dual-Port Memory Interface, Revision 17, English, 2020.
[2]	Hilscher Gesellschaft für Systemautomation mbH: Packet API, netX Dual-Port Memory, Packet-based services (netX 90/4000/4100), Revision 5, English, 2020.
[3]	Hilscher Gesellschaft für Systemautomation mbH: Authentication Manager, Certificate Database, Revision X, English, 2022.

Table 2. References to Documents

Chapter 2 Hilscher General Firmware with User Management

2.1 Structure of the Hilscher Firmware with User Management

The figure below shows the internal structure of a Hilscher LFW with User Management features. The LFW consists of the RTE protocol stack (green highlighted components), optional Network Services (yellow highlighted components) and the User Management relevant components (red highlighted components).

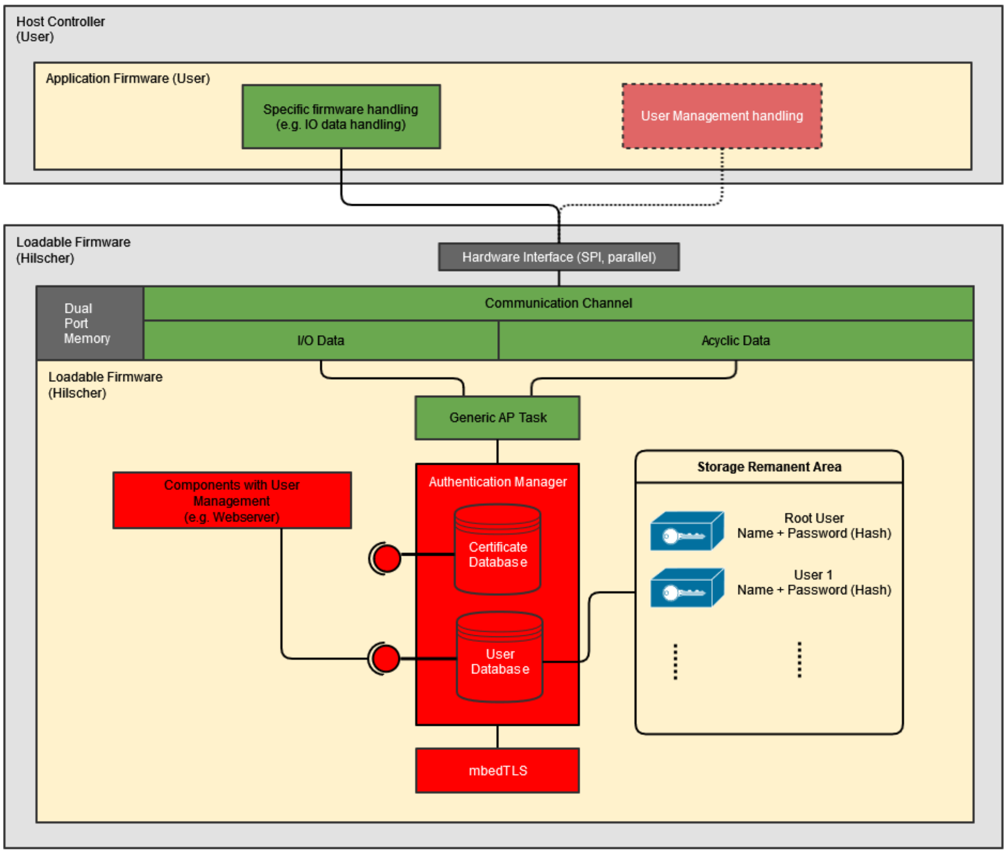

Figure 1. Hilscher Security Firmware Structure

In the following only the User Management related components are briefly described.

The **Authentication Manager** provides the User Database API for user access the user management and the Certificate Database API for key and certificate management [3]. The services are available via DPM communication channel 0.

The **mbedTLS** is a lightweight open source cryptography and security library which provides Hash functionalities to encrypt the user information stored in the User Database.

2.2 User Database Resources Organization

The following component diagram explains the resources available for each User in the User Database.

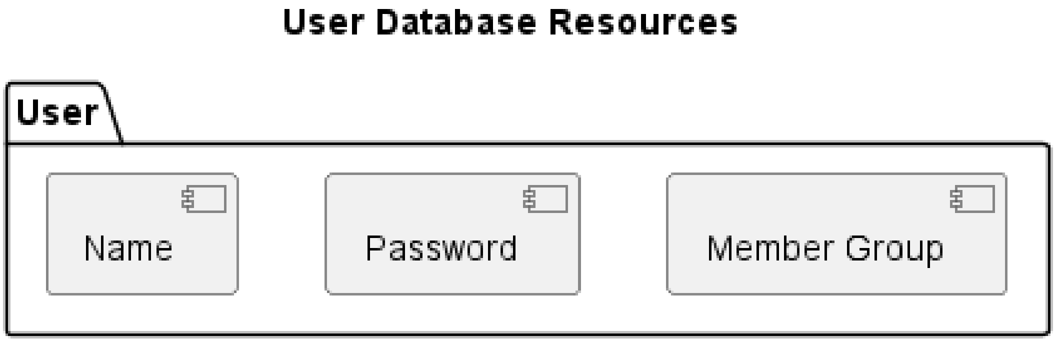

Figure 2. User Database Resources

The particular User Database resource is described in the following table.

Resource type	Description
Name	Each user is identified by its name. Thus, it must be unique in the database.
Password	The password is used to authenticate the user and it is stored in a Hash container into the User Database.
Member Group	This attribute defines, to which group(s) the user belongs to (e.g. Admin, Guest, Account Manager, etc)

Table 3. User Database Resource Description

2.3 Default Root User

The Default Root User is the first user added to the User Database during the firmware initialization. The user name and passwords are set via the startup parameters of the User Database component during the LFW startup. The common Hilscher LFW uses the following Root User: - Name: "root" - Password: "password"

NOTE The Root User may be different in some products. In this case, please refer to the description of the respective LFW in order to obtain the specific Root User name and password.

NOTE It is highly recommended to change the default root user name and password in the final product.

The Default Root User can be changed by the application using the Application Interface. Please refer to [sec-userdb-usecase-configureroot] for more details.

2.4 Privileged / non-privileged API

Some User Database services change the content of the database (e.g. add/delete user). Thus, these services are declared as "privileged" and can only be triggered by an user from the groups "Admin" or "Account Manager".

Other services don’t modify the User Database content (e.g. authenticate user), Therefore, they are declared as "non-privileged" and can be triggered without restrictions.

Chapter 3 Getting Started/Configuration

3.1 Change Root User

The following activity diagram shows how an application can replace the LFW Default Root User. Typically, this step has to be done in the following scenarios: - First initialization after out-of-box - After a Reset To Factory Defaults was performed

NOTE Please refer to the description of the respective LFW in order to obtain the Default Root User name and password.

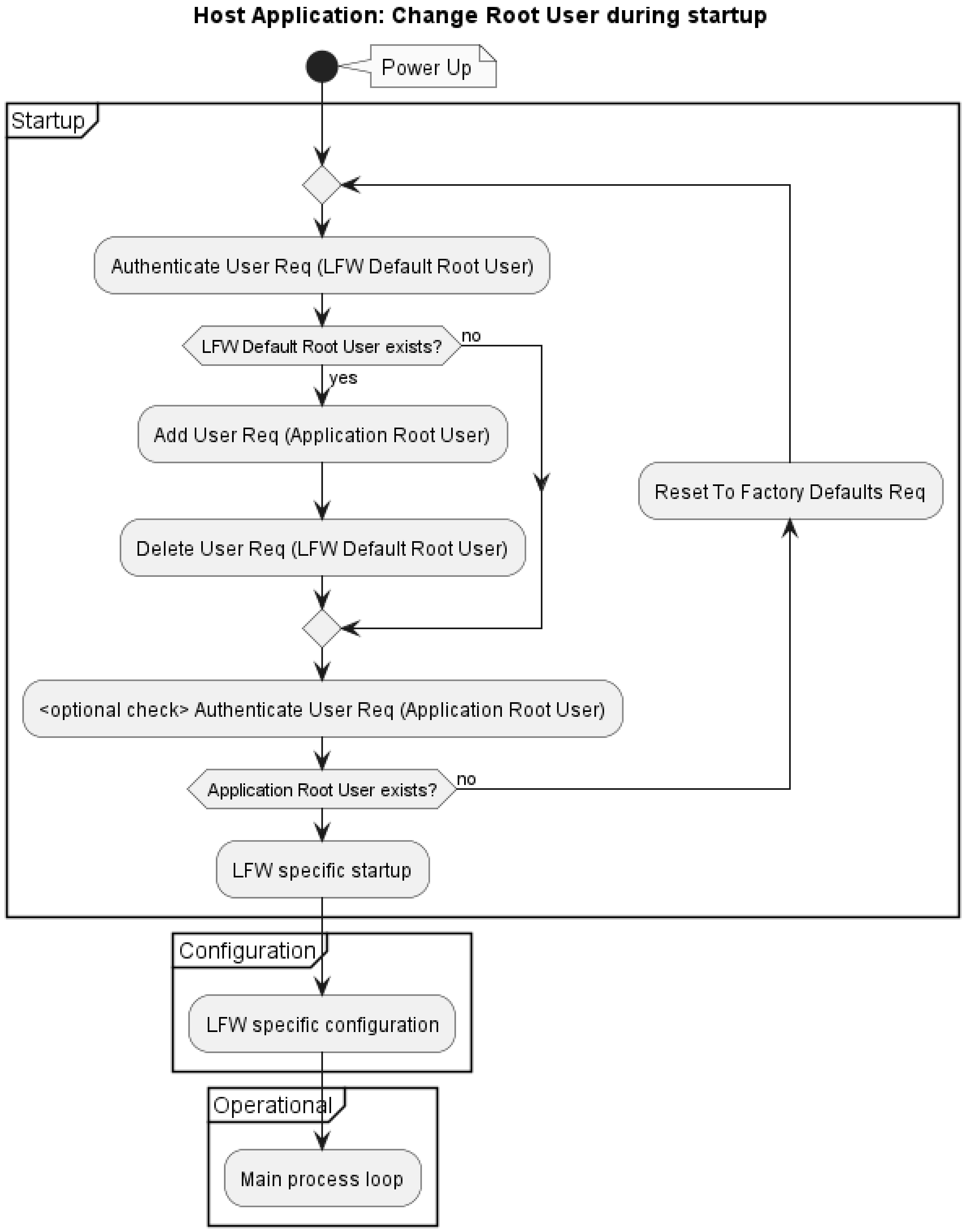
Host Application: Change Root User during startup
Power Up
Startup
Authenticate User Req (LFW Default Root User)
LFW Default Root User exists?
no
yes
Add User Req (Application Root User)
Delete User Req (LFW Default Root User)
Reset To Factory Defaults Req
<optional check> Authenticate User Req (Application Root User)
Application Root User exists?
no
LFW specific startup
Configuration
LFW specific configuration
Operational
Main process loop

Figure 3. Host Application: Change Root User during startup

3.2 Configuration packets

Packet name	Definition	Command Code
Authenticate User	AUTHMGR_USRDB_AUTHENTICATE_USER_REQ	0xB000/0xB001
Add User	AUTHMGR_USRDB_ADD_USER_REQ	0xB002/0xB003
Delete User	AUTHMGR_USRDB_DELETE_USER_REQ	0xB004/0xB005
Change Password	AUTHMGR_USRDB_CHANGE_PASSWORD_REQ	0xB006/0xB007
Reset to Factory Defaults	AUTHMGR_USRDB_RESET_TO_FACTORY_DEFAULTS_REQ	0xB008/0xB009

Table 4. User Database Configuration Packets

Chapter 4 Application Interface

This chapter defines the application interface of the Authentication Manager. User Database

4.1 User Management Resources

4.1.1 User Groups

Value	Name Description
0x00000001	MSK_AUTH_USRDB_GROUP_GUEST User group "guest"
0x00000002	MSK_AUTH_USRDB_GROUP_USER User group "user"
0x00000004	MSK_AUTH_USRDB_GROUP_MANAGER User group "manager"
0x00000008	MSK_AUTH_USRDB_GROUP_FWUPDATE User group "firmware update"
0x00000010	MSK_AUTH_USRDB_GROUP_RESET User group "reset"
0x00000100	MSK_AUTH_USRDB_GROUP_PRODUCT_SPECIFIC_0 User group "product specific 0"
0x00000200	MSK_AUTH_USRDB_GROUP_PRODUCT_SPECIFIC_1 User group "product specific 1"
0x00000400	MSK_AUTH_USRDB_GROUP_PRODUCT_SPECIFIC_2 User group "product specific 2"
0x00000800	MSK_AUTH_USRDB_GROUP_PRODUCT_SPECIFIC_3 User group "product specific 3"
0x00001000	MSK_AUTH_USRDB_GROUP_PRODUCT_SPECIFIC_4 User group "product specific 4"
0x00002000	MSK_AUTH_USRDB_GROUP_PRODUCT_SPECIFIC_5 User group "product specific 5"
0x00004000	MSK_AUTH_USRDB_GROUP_PRODUCT_SPECIFIC_6 User group "product specific 6"
0x00008000	MSK_AUTH_USRDB_GROUP_PRODUCT_SPECIFIC_7 User group "product specific 7"
0x04000000	MSK_AUTH_USRDB_GROUP_USER_ACCOUNT_MANAGER User group "user account manager"
0x80000000	MSK_AUTH_USRDB_GROUP_ADMIN User group "admin"

Table 5. AUTH_USRDB_USER_GROUP_BF_T

4.1.2 User Structure

AUTHMGR_USRDB_USER_T		
Variable	**Type**	**Description**
szUserName	char[28]	User name (non-empty, '\0' terminated string)
szPassword	char[28]	User password (non-empty, '\0' terminated string)

Table 6. AUTHMGR_USRDB_USER_T

4.1.3 Resources limits

4.1.3.1 Maximum User Name Length.

AUTHMGR_USRDB_USER_NAME_LEN_MAX	28

4.1.3.2 Maximum User Password Length.

AUTHMGR_USRDB_PASSWORD_LEN_MAX	28

4.2 Authenticate User

Authenticate User Command Request

AUTHMGR_USRDB_AUTHENTICATE_USER_REQ	0x0000B000

Authenticate User Command Request

AUTHMGR_USRDB_AUTHENTICATE_USER_CNF	0x0000B001

Authenticate User Request Packet Description

Authenticate user with user name and password.

Variable	Type	Description
tHead	**HIL_PACKET_HEADER_T**	
`ulDest`	uint32_t	
`ulLen`	uint32_t	56
`ulSta`	uint32_t	0
`ulCmd`	uint32_t	AUTHMGR_USRDB_AUTHENTICATE_USER_REQ
tData	**AUTHMGR_USRDB_AUTHENTICATE_USER_REQ_DATA_T**	
`tUser`	AUTHMGR_USRDB_USER_T	User to authenticate (see AUTHMGR_USRDB_USER_T)

Table 7. AUTHMGR_USRDB_AUTHENTICATE_USER_REQ_T

Authenticate User Confirmation Packet Description

Variable	Type	Description
tHead	**HIL_PACKET_HEADER_T**	
`ulDest`	uint32_t	
`ulLen`	uint32_t	4
`ulSta`	uint32_t	0
`ulCmd`	uint32_t	AUTHMGR_USRDB_AUTHENTICATE_USER_CNF
tData	**AUTHMGR_USRDB_AUTHENTICATE_USER_CNF_DATA_T**	
`ulGroupMember`	AUTH_USRDB_USER_GROUP_BF_T	Groups the user belongs to (see AUTH_USRDB_USER_GROUP_BF_T)

Table 8. AUTHMGR_USRDB_AUTHENTICATE_USER_CNF_T

4.3 Add User

Add User Command Request

AUTHMGR_USRDB_ADD_USER_REQ	0x0000B002

Add User Command Request

AUTHMGR_USRDB_ADD_USER_CNF	0x0000B003

Add User Request Packet Description

Add new user to the User Database

NOTE This is a privileged service and can only be triggered by an user from the groups "Admin" or "Account Manager".

NOTE It is disallowed to add an user which belongs to a higher group than the privileged user

Variable	Type	Description
tHead	**HIL_PACKET_HEADER_T**	
ulDest	uint32_t	
ulLen	uint32_t	116
ulSta	uint32_t	0
ulCmd	uint32_t	AUTHMGR_USRDB_ADD_USER_REQ
tData	**AUTHMGR_USRDB_ADD_USER_REQ_DATA_T**	
tUserPrivileged	AUTHMGR_USRDB_USER_T	User that is privileged to create the following user (see AUTHMGR_USRDB_USER_T)
tUserAdd	AUTHMGR_USRDB_USER_T	New user to be created (see AUTHMGR_USRDB_USER_T)
ulGroupMember	AUTH_USRDB_USER_GROUP_BF_T	Groups the user belongs to (see AUTH_USRDB_USER_GROUP_BF_T)

Table 9. AUTHMGR_USRDB_ADD_USER_REQ_T

Add User Confirmation Packet Description

Variable	Type	Description
tHead	**HIL_PACKET_HEADER_T**	
ulDest	uint32_t	
ulLen	uint32_t	0
ulSta	uint32_t	0
ulCmd	uint32_t	AUTHMGR_USRDB_ADD_USER_CNF

Table 10. AUTHMGR_USRDB_ADD_USER_CNF_T

4.4 Delete User

Delete User Command Request

AUTHMGR_USRDB_DELETE_USER_REQ	0x0000B004

Delete User Command Request

AUTHMGR_USRDB_DELETE_USER_CNF	0x0000B005

Delete User Request Packet Description

Delete user from the User Database.

NOTE This is a privileged service and can only be triggered by an user from the groups "Admin" or "Account Manager".

NOTE It is disallowed to delete an user which belongs to a higher group than the privileged user

Variable	Type	Description
tHead	**HIL_PACKET_HEADER_T**	
`ulDest`	uint32_t	
`ulLen`	uint32_t	84
`ulSta`	uint32_t	0
`ulCmd`	uint32_t	AUTHMGR_USRDB_DELETE_USER_REQ
tData	**AUTHMGR_USRDB_DELETE_USER_REQ_DATA_T**	
`tUserPrivileged`	AUTHMGR_USRDB_USER_T	User that is privileged to delete the following user (see AUTHMGR_USRDB_USER_T)
`szUserName[AUTHMGR_USRDB_USER_NAME_LEN_MAX]`	char	Name of the user to be deleted (non-empty, '\0' terminated string)

Table 11. AUTHMGR_USRDB_DELETE_USER_REQ_T

Delete User Confirmation Packet Description

Variable	Type	Description
tHead	**HIL_PACKET_HEADER_T**	
`ulDest`	uint32_t	
`ulLen`	uint32_t	0
`ulSta`	uint32_t	0
`ulCmd`	uint32_t	AUTHMGR_USRDB_DELETE_USER_CNF

Table 12. AUTHMGR_USRDB_DELETE_USER_CNF_T

4.5 Change Password

Change Password Command Request

AUTHMGR_USRDB_CHANGE_PASSWORD_REQ	0x0000B006

Change Password Command Request

AUTHMGR_USRDB_CHANGE_PASSWORD_CNF	0x0000B007

Change Password Request Packet Description

Change the user password.

Variable	Type	Description
tHead	**HIL_PACKET_HEADER_T**	
`ulDest`	uint32_t	
`ulLen`	uint32_t	84
`ulSta`	uint32_t	0
`ulCmd`	uint32_t	AUTHMGR_USRDB_CHANGE_PASSWORD_REQ
tData	**AUTHMGR_USRDB_CHANGE_PASSWORD_REQ_DATA_T**	
`tUser`	AUTHMGR_USRDB_USER_T	Existing user for whom the password shall be changed (see AUTHMGR_USRDB_USER_T)
`szPasswordNew[AUTHMGR_USRDB_PASSWORD_LEN_MAX]`	char	New user password (non-empty, '\0' terminated string)

Table 13. AUTHMGR_USRDB_CHANGE_PASSWORD_REQ_T

Change Password Confirmation Packet Description

Variable	Type	Description
tHead	**HIL_PACKET_HEADER_T**	
`ulDest`	uint32_t	
`ulLen`	uint32_t	0
`ulSta`	uint32_t	0
`ulCmd`	uint32_t	AUTHMGR_USRDB_CHANGE_PASSWORD_CNF

Table 14. AUTHMGR_USRDB_CHANGE_PASSWORD_CNF_T

4.6 Reset to Factory Defaults

Reset to Factory Defaults Command Request

AUTHMGR_USRDB_RESET_TO_FACTORY_DEFAULTS_REQ	0x0000B008

Reset to Factory Defaults Command Request

AUTHMGR_USRDB_RESET_TO_FACTORY_DEFAULTS_CNF	0x0000B009

Reset to Factory Defaults Request Packet Description

Reset the User Database to its Factory Defaults. The complete user database is deleted and the root user is re-created from the configuration passed during firmware initialization. The component remains initialized.

> **NOTE** This is a privileged service and can only be triggered by an user from the groups "Admin" or "Account Manager".

> **NOTE** After the reset, the pre-compiled LFW Default Root User is re-created. If the application wants to replace it by its own Root User, this has to be done separately

Variable	Type	Description
tHead	**HIL_PACKET_HEADER_T**	
`ulDest`	uint32_t	
`ulLen`	uint32_t	56
`ulSta`	uint32_t	0
`ulCmd`	uint32_t	AUTHMGR_USRDB_RESET_TO_FACTORY_DEFAULTS_REQ
tData	**AUTHMGR_USRDB_RESET_TO_FACTORY_DEFAULTS_REQ_DATA_T**	
`tUserPrivileged`	AUTHMGR_USRDB_USER_T	User that is privileged to trigger the reset to factory defaults (see AUTHMGR_USRDB_USER_T)

Table 15. AUTHMGR_USRDB_RESET_TO_FACTORY_DEFAULTS_REQ_T

Reset to Factory Defaults Confirmation Packet Description

Variable	Type	Description
tHead	**HIL_PACKET_HEADER_T**	
`ulDest`	uint32_t	
`ulLen`	uint32_t	0
`ulSta`	uint32_t	0
`ulCmd`	uint32_t	AUTHMGR_USRDB_RESET_TO_FACTORY_DEFAULTS_CNF

Table 16. AUTHMGR_USRDB_RESET_TO_FACTORY_DEFAULTS_CNF_T