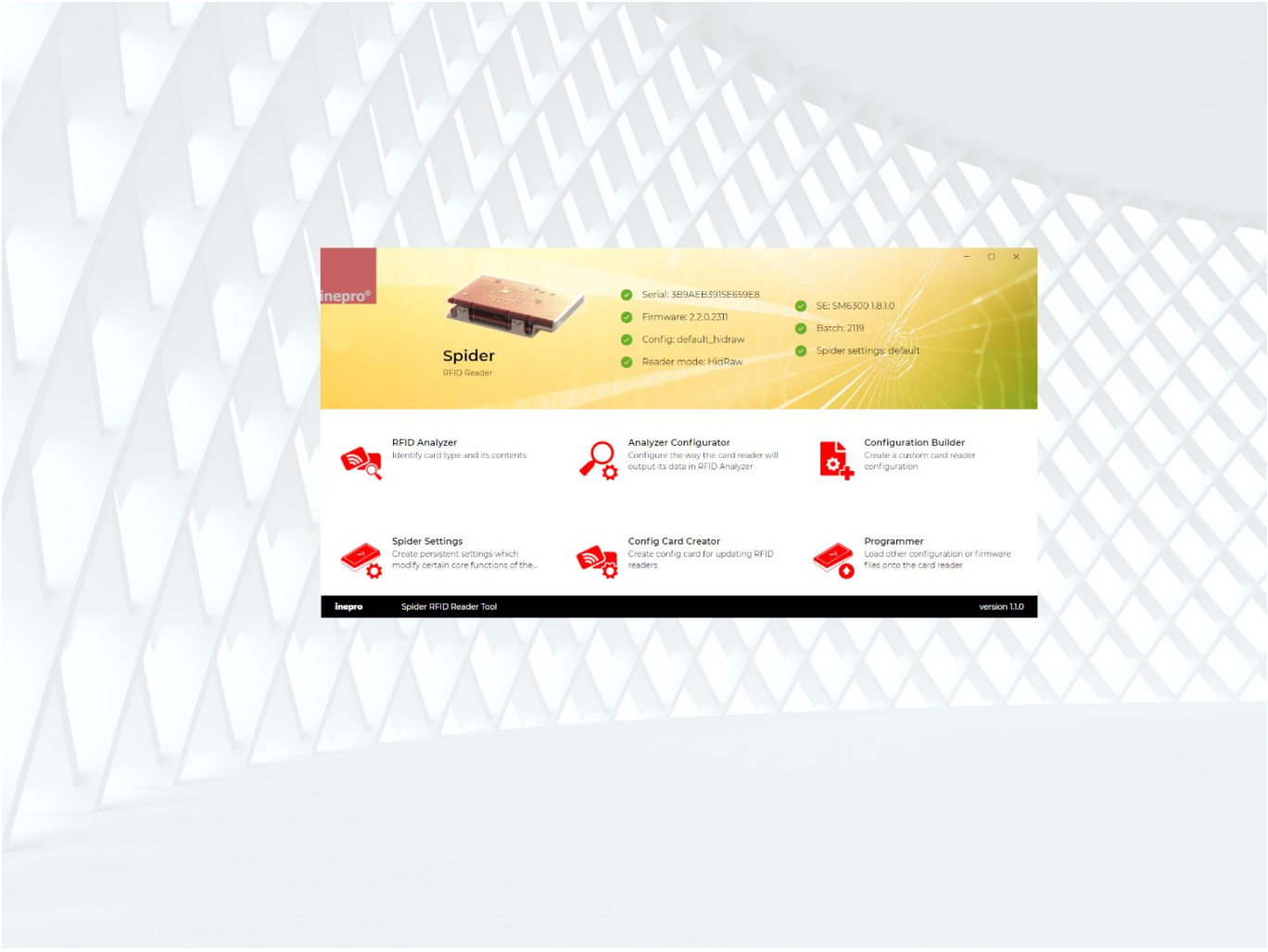
Spider RFID Reader Tool



Congratulations on your selection of the Spider RFID Reader Tool. We are certain you will be pleased with your purchase of one of the flexibele solutions of the market.

We want to help you get the best result from your Spider RFID Reader Tool. This manual contains information on how to do that; please read it carefully. Due to continuous product improvements this manual is subject to changes without notice.

We strongly recommend you read the license agreement to fully understand its coverage and your responsibilities of ownership.

Your dealer is dedicated to your satisfaction and will be pleased to answer your questions and your concerns.

Best wishes,

inepro B.V.

inepro®

Spider RFID Reader Tool - Instruction Manual | EN

Revision number R 10. This revision of the manual can be applied to products with version 1.3.0 or higher.

Publisher
inepro B.V.

Managing Editor
K. de Graaf

Technical Editors
T. Tjie
K. de Graaf

Cover Design
K. de Graaf
M. Sterk

Team Coordinator
R. Groen

Production
inepro B.V.

Spider RFID Reader Tool

Table of Contents

Introduction

The Spider RFID Reader Tool can configure the Spider RFID Reader, the SCR708 and the Red Spider (to be released) in a number of ways. It can upload firmware or configuration files to the Spider RFID Reader and last but not least it can analyse cards, identifying the card type and data that is on the card. In this manual only the Spider will be mentioned to illustrate how the tool works for inepro readers.

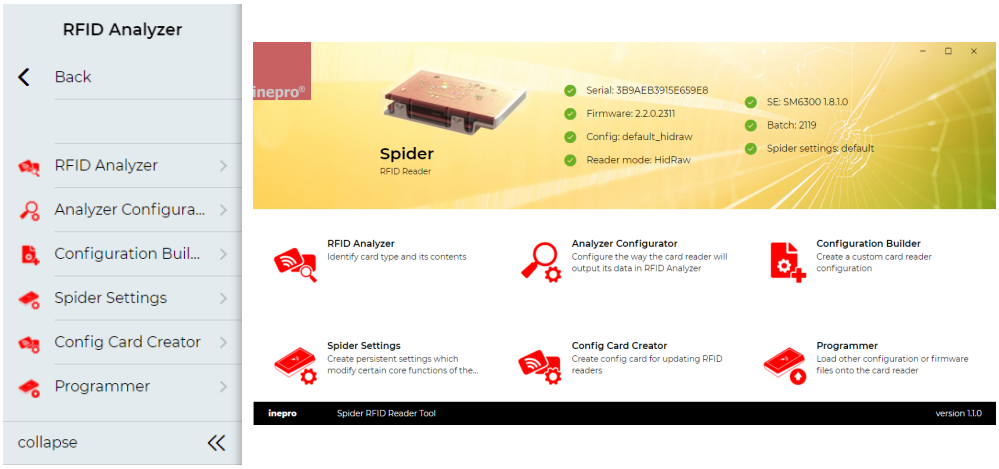
You can tweak the RFID Analyser configuration with the Analyzer Configurator, set the persistent settings in the Spider RFID Reader at Spider Settings, create a Config Card to transfer a configuration to multiple Spider RFID Readers, build your own custom configuration with the Configuration Builder or use the Programmer to load a configuration and/or firmware on the attached Spider RFID Reader.

We wish you success in your Spider RFID endeavours and hope you like our Spider RFID Tool,

The inepro team ■

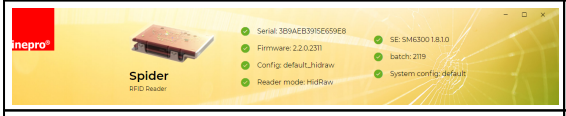
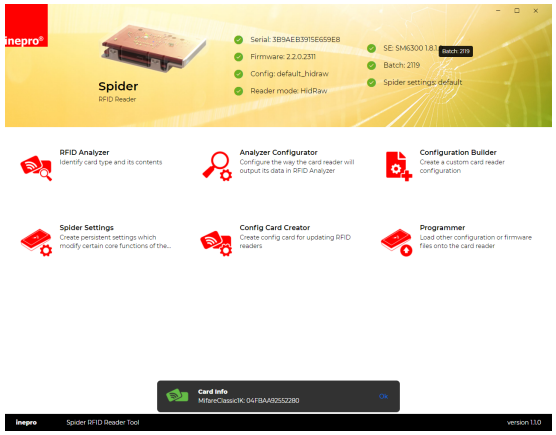
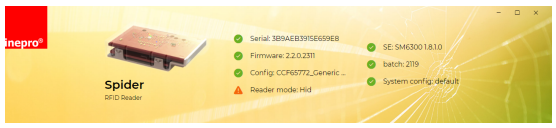
Dashboard

The main window of the Spider RFID Tool has all it's different processes in one place. At the top left corner, it will show the type of reader (Spider, Red Spider or SCR708) that has been connected. The menu or Dashboard is also available in the rest of the app as a menu bar on the left side of the window. Apart from that it will show the Spider RFID reader's status on the top bar.



Statuses

The Spider RFID Reader can show all kind of info in it's status screen, the most common are explained here.

	The Spider RFID Reader is disconnected
	Spider RFID Reader is connected, Reader mode is HidRaw (Note: Card analysis is only possible in HID RAW Mode)
	Spider RFID Reader is connected, Reader mode is HidRaw, Card info of the most recent placed card is MifareCassic1K | 04FBAA92552280
	Spider RFID Reader is connected, Reader mode is not HidRaw (but Hid) (Note: Card analysis is not possible, as shown by the orange triangle)

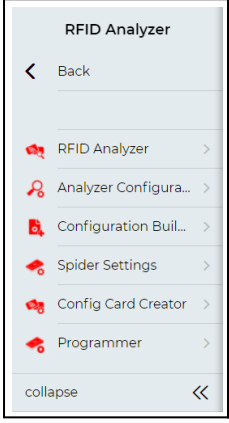
Dashboard

Other info

In the status window you can also find other information about the Spider RFID Reader.

	1. Serial number 2. Firmware version 3. Configuration file loaded 4. Reader Mode 5. Secure Element Chip | Type and version 6. Product batch number 7. Spider Settings Configuration Name

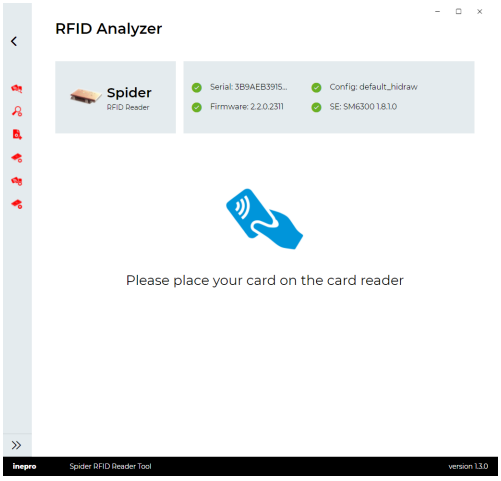
Left Menu bar

	When you leave the dashboard, the left menu bar will be shown. It has the title of the current window (replaced here with the placeholder '<title>'). Under that you will find the option to go Back to the dashboard (click 'Back') and a list of links to the other components of the Spider RFID Reader Tool ■

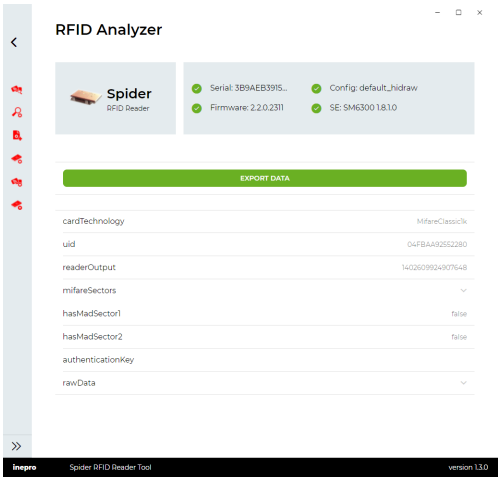
RFID Analyzer

The RFID Analyzer analyzes the RFID tag or card and will show you every piece of data that it finds on the RFID.
The UID and the Reader Output are the most important values as they are respectively the 'unique' ID of the card and the derivative ID that can be formatted to give the ID format that is needed.
It will also provide the detected card technology (note that some card technologies overlap / are interchangeable and can therefore be detected as the one another. To prevent this the unused card technologies can be disabled in the Spider RFID reader so that they will not be detected).

1. Place the card (or other RFID credential) on the reader. Approach the reader gently to give the reader time to read the card at it's ideal range. It may take a while (max. 1 min.) to recognize the RFID credential. If it takes longer and it is still not recognized, take the card off and wait until you only see the blue LED flicker and repeat the card placing procedure.



2. The reader will beep and the card data will be shown on the screen.
3. Explore the data to find what you need. Is the reader output what you would expect to be the card's ID? Interested in the other information? Note that a number of menu's can be expanded, like the sector details.



4. If the Reader Output is what is expected please continue to check another card until you have confirmed with at least three cards that the Reader Output is the expected card ID.
5. If the Reader Output is not what is expected, please go to the Analyzer Configurator to make simple changes to the card ID format and find the expected format ■

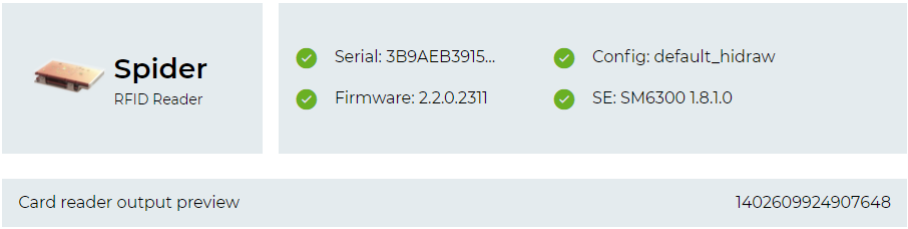
Analyzer Configurator

In the Analyzer Configurator you can create a quick configuration to use in the RFID Analyzer without having to store it anywhere, although storing it is possible. The configuration will effect the way the Spider RFID Reader will output its data in RFID analyzer.

It is used to adapt the Card Reader Output format in such a way that the Card Reader Output produces the expected card ID in the correct format. That format is then saved and can be programmed into your reader using the Programmer 22.

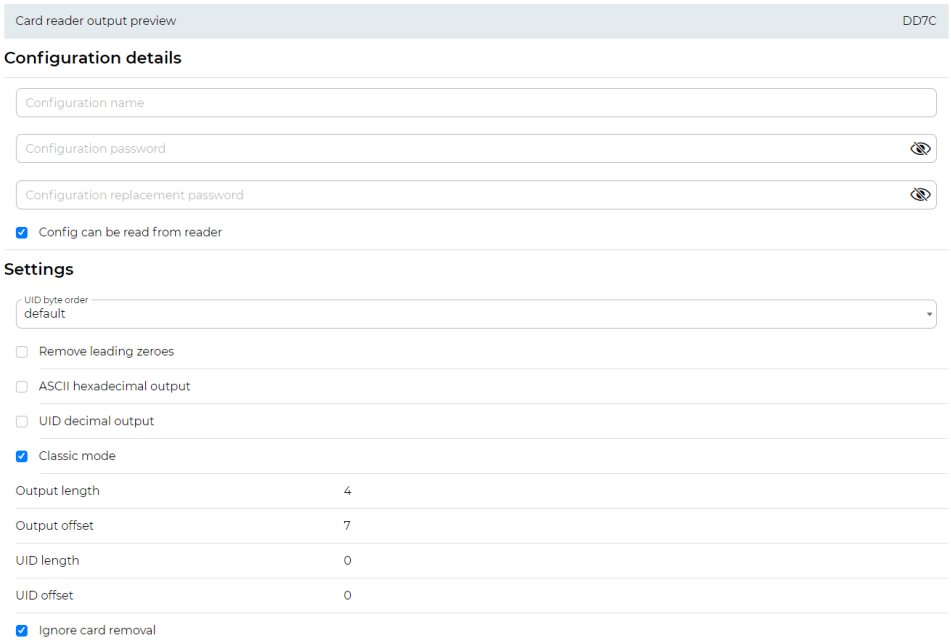
How to get my Card Reader Output format?

To get the expected Card Reader Output format, you need to have a clear image what format you need.
Make sure to have at least three sample card for which the expected outcome is known.
Use each of those cards to test if the format works as expected.



Say we start with the hexdecimal version of the UID in the example above; '04FBAA92552280' and wanted only the 'BAA9' section.

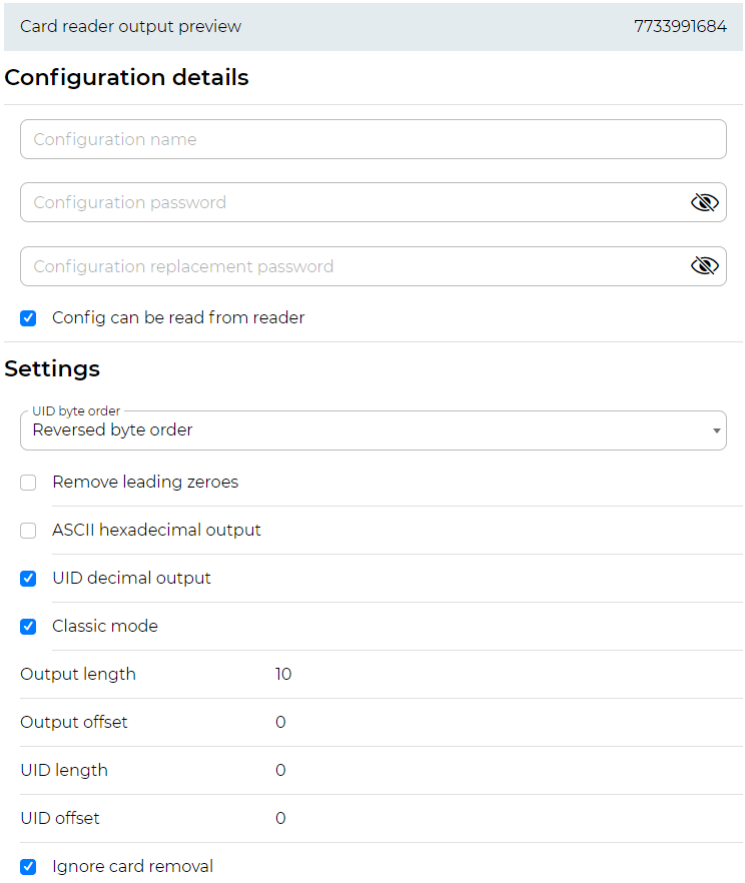
1. Classic mode should be on, and the card reader output preview should give us a decimal numeric ID (in this example 1402609924907648). Turn off UID decimal Output to show the hexadecimal 'BAA9' section.



2. We would first set the '**Output offset**' to 7 to cut off the right side, this renders us '04FBAA9'.
3. Then we would set the '**Output length**' to be 4 to cut off the left side, this leave us with the wanted format 'BAA9'

Let's try this exercise again. Say we need a 10 digit decimal number (so no hex values) and we know that the number (of this particular card) needs to start with 77.

1. We would first set our output to '**UID decimal output**', this renders us '1264265541526144', but no '77' is found in that output!?
2. What we could do is try the enable the '**ASCII hexadecimal output**'. Alas, this gives us '31323634323635353431353236313434' which has a lot of '3's but no '77'.
3. Disable the 'ASCII hexadecimal output' again and now try to set the '**UID byte order**' to '**reverse**' as opposed to '**default**', with the result '36066737733991684', we now do have a '77' and can now cut our result output to the correct length of 10.
4. Set '**Output length**' to '10'.



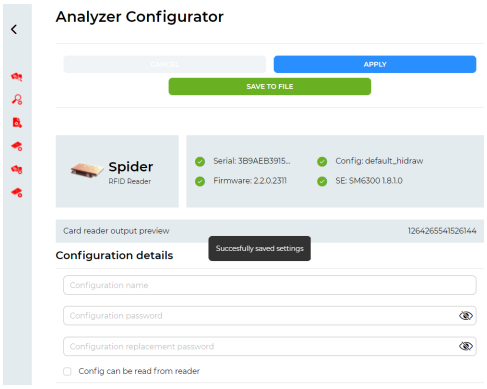
We have now converted two UID's to another format to get the wanted result. Use this process to find the correct Card Reader Output for your RFID credentials (card, tags or other). Refer to the table below for more information on each setting.

Configuration Settings

Name	Default Value	Description
UID byte order	default	Can be set to 'default' or 'reversed byte order' to reverse the byte order
- Remove leading zeroes	false	If set, it will remove the zeroes at the front of the number value, for example '00123' => '123'
- ASCII hexadecimal output	false	Each character in the UID has an hex code in the ASCII table. this option converts each character to its hex ASCII code, making the code double as long, as each hex code is two digits. For example '0' => '30' and so the full ID '047DD7C2812280' => '30343744443743323831323238330'.
- UID decimal output	true	If set, it will output the value as decimal, for example '0A7F' => '2687'
- Classic mode	true	Classic Mode to true will render a format that only contains the card number. Classic Mode to false will render a comma separated format that also includes the incoming/outgoing card direction, the Card Type and an inepro specific identifier.
Output length	0	If set, the output length will be restricted to that length, for example if length is 15; '047DD7C2812280' => '0047DD7C2812280' => . 0 will render no restrictions.
Output offset	0	If set, the output will be offset by that number, losing the characters on the right side, for example if offset is 3; '047DD7C2812280' => '047DD7C2812'. 0 will render no offset.
UID length	0	If set, the output byte length will be set to that length, this may drastically change the output, for example if length is 8; '047DD7C2812280' => '00047DD7C2812280'. 0 will render no restrictions.
UID offset	0	If set, the output will be offset by that number of bytes, losing bytes on the rights side for example if offset is 3; '047DD7C2812280' => '047DD7C2'. 0 will render no offset.
Ignore card removal	true	If set to true it will no longer react to the 'card-removed' event, (no card data or enter will be sent as output).

Apply the settings for the analyser

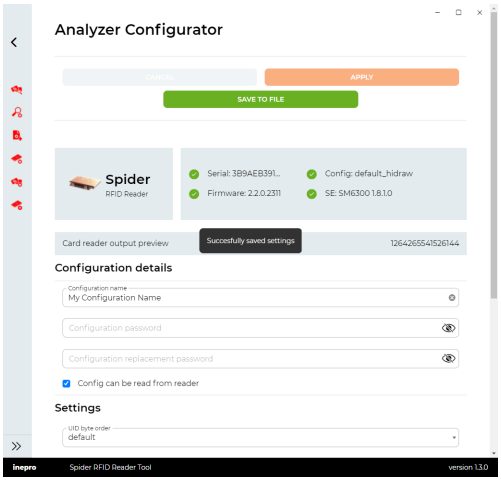
Click on 'Apply' to apply the settings, they can now be used in the Analyser.



Found the correct Card Reader Output format!

You have successfully explored the settings and managed to find a suitable Card Reader Output Format. You would now like to store this configuration in the reader you it will always use this format to read the cards (or other RFID credentials). This is very simple:

1. Give the configuration a name.
2. Optionally secure it with a password (configurations cannot be viewed nor edited without this password).
3. Click 'Save to File' and browse to folder and give it a file name.
4. Click on 'Save' to confirm in the 'Save File' dialog
5. Then go to the Programmer 22 (in the left menu bar) to program the reader with this file ■

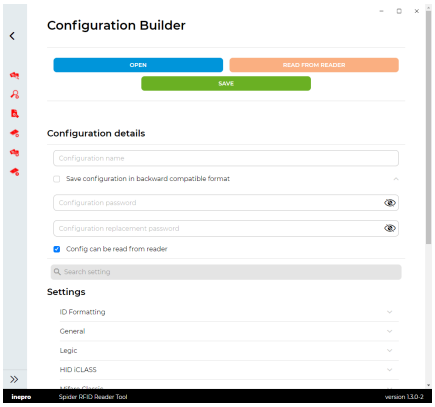


Configuration Builder

The Configuration Builder can be used to either create or modify a custom Spider RFID Reader configuration or to read one out.

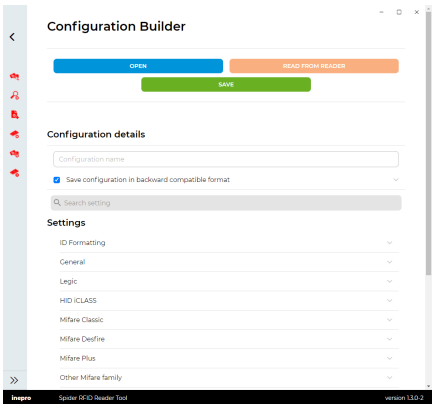
Create a new Spider RFID Reader configuration file

1. Give you configuration a fitting and unique name.

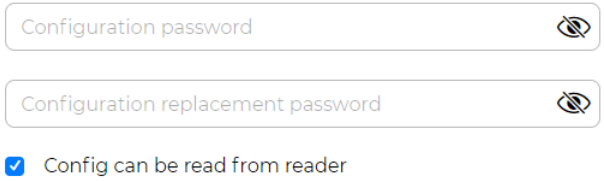


2. If you use Inepro reader models previous to the Spider like the SCR708 in your project, make sure the tick the 'Save configuration in backward compatible format' box.

The backwards compatible version can be edited in the older editor and therefore will not be able to be protected with a password.

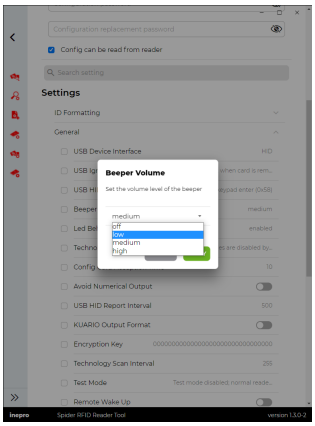


3. **Optional** | Set a optionally configuration password to make sure unauthorised people are unable to open or edit it.



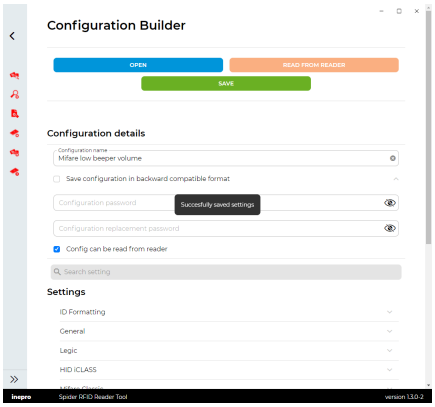
4. It is recommended to leave the 'Config can be read from reader' ticked for your convenience, but for enhanced security it can be disabled. In that case the configuration file can not be extracted from the Spider RFID Reader.

5. Adapt the settings until the configuration is what you need.



6. Save the Configuration, making sure you gave it an unique recognisable name.

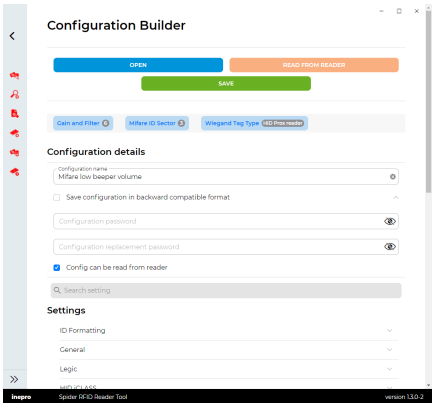
A message will appear to confirm that the configuration has been saved.



Configuration Builder

Load and read an existing Spider RFID Reader configuration file

1. Click open and open a valid configuration file.

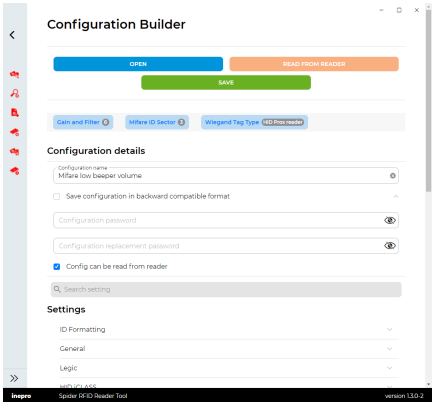


All values in the file will be presented in the tool, if a setting section has been changed it will be indicated with a badge at the category carrying the number of changes inside that category. On the top you will see badges with all the effected settings by name and with their current value.

These badges link directly to those settings, so you can click and edit them directly.

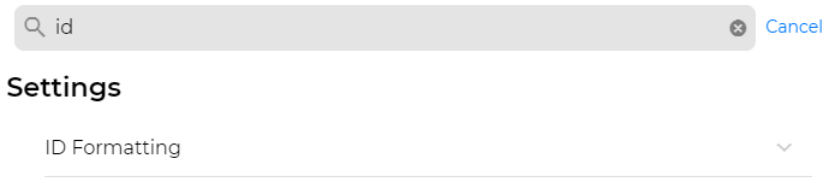
Load a file directly from a reader

1. Click on "read from reader" and the configuration currently in the reader will be loaded.



Search settings

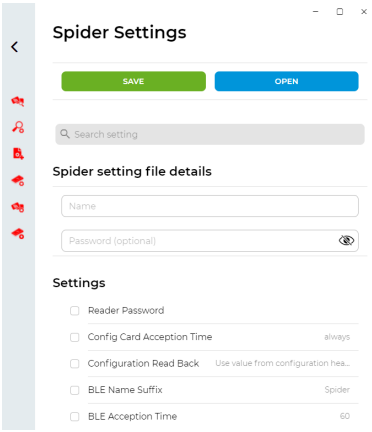
In the 'Search setting' bar you can search for a setting name in case you can't find it in the category list.



The search results will be sorted by category ■

Spider Settings

The Spider Settings are persistent settings which modify certain device properties of the Spider RFID reader.'
'Persistent' in our case means that when a new set of Spider Settings is loaded, only settings that are explicitly overwritten will change, unchanged settings will remain what they have been set to before. In other words, loading a new set of Spider Settings will not purge the old set of Spider Settings within the Spider RFID Reader, rather they will be merged with each other.



Spider Settings

Spider settings file details

The set of Spider Settings should have a distinct name. We can name them in the Configuration Details section at 'Configuration Name'. This name is mandatory and is used on the Dashboard when the reader is connected to the Spider RFID Reader Tool to show you which Spider Settings file is loaded last in your Spider RFID Reader.

A Configuration Name cannot be longer then 30 characters!

The settings can be protected with a password. Without the password the Spider Settings file cannot be displayed.
Future settings might enforce the use of a password.

These Spider Settings apply only to the Spider RFID Reader and can only be stored as an *.advancedreaderconfig file

Spider Settings

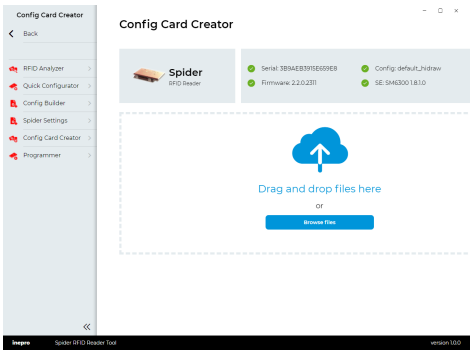
Name	Default Value	Description
Reader Password	<empty>	A spider RFID reader can be password protected. This password is needed to protect administrative access with the BLE Configurator App to the Spider RFID reader. For example to change it's configuration. At the moment of writing the BLE Configurator App cannot be used to gain access to a password-protected Spider RFID Reader. In a future update the BLE Configurator App will have the ability to prompt for a configuration password, allowing access to password-protected Spider RFID readers. If there is no reader password set, access is possible either by using the Spider RFID Reader Tool with USB connection or by using the the BLE Configurator App and scanning the QR code belonging the Spider RFID Reader. That QR can be found on the serial sticker.
Configuration Card Acceptance Time	30	Time in seconds that configuration cards are recognised after power-up. Valid values are: never | **30** | 60 | 120 | 240 | always

Spider Settings

Name	Default Value	Description
Configuration Read Back	Use value from configuration header	The Spider RFID Reader is capable of reading it's entire configuration out of it's memory to the Spider RFID Reader Tool. It is possible to disable this function. This can be done by a normal configuration file or is overruled by this persistent Spider Setting. When using **Always Allowed** the value in the configuration file is overruled, and reading out the configuration will always be possible. The **disabled** value will never allow this reader to read back the configuration file. Valid values are: disabled | **Use value from configuration header** | Always allowed.
BLE Name Suffix	"Spider"	The 'friendly' name of the Spider RFID Reader's BLE ID in BLE apps. This name can be maximum 16 characters long. We recommend to use a name that identifies the location and/or device that the reader is at. This will help users and administrators to identify the correct Spider RFID Reader. Alternatively this setting can be configured through the Spider Configurator app.
BLE Acceptance Time	60	Time in seconds that administrative access over BLE is allowed after power-up. This setting does not effect the Spider ID app Valid values are: never | 30 | **60** | 120 | 240 | always

Config Card Creator

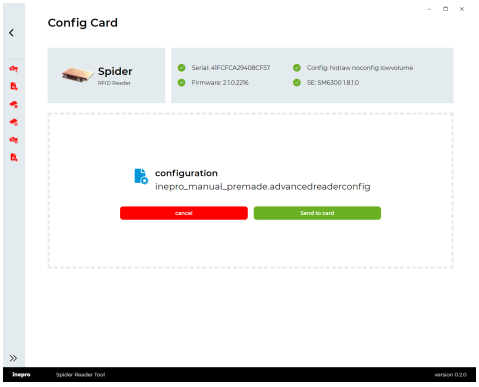
Create configuration card(s) to update Spider RFID Readers with a specific configuration.



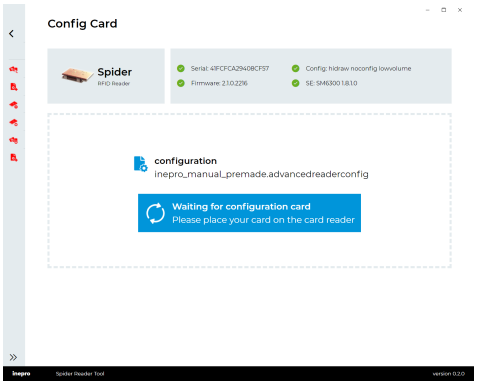
You need a new (unused) DESFire card and the desired configuration file to start this process!
Used DESFire cards cannot be used again in this process.

The configured card will be used to program Spider RFID readers with the written configuration file by presenting the card to the Spider RFID reader within a specific time from the boot moment (as specified in the Spider Setting Configuration Card Acceptence Time 18).

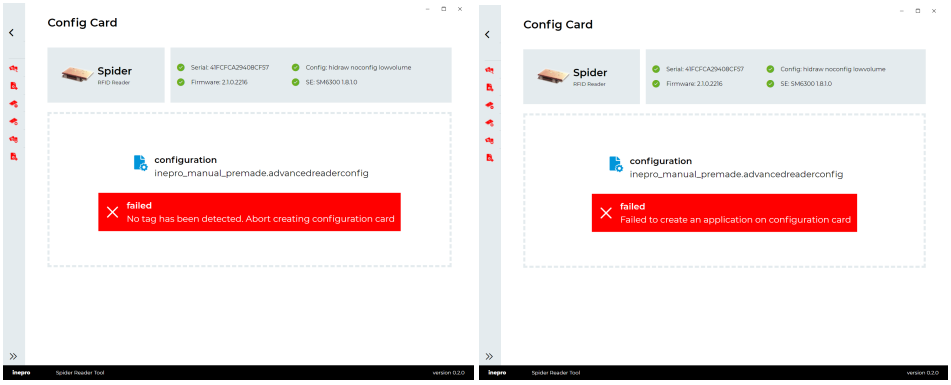
1. Either drag and drop the file into the drag-and-drop-area or click 'Browse Files' to browse for a valid configuration file.



2. Configuration file selected. Click on “send to card” to continue.

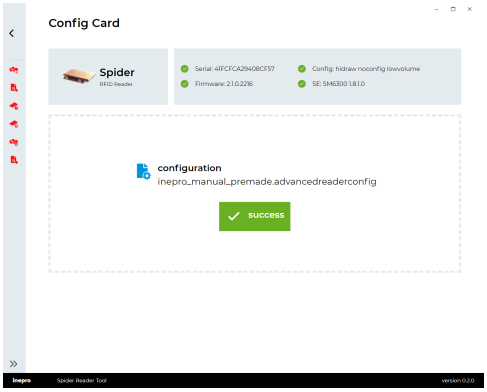


3. Lay an empty DESFire card on the Spider RFID Reader.



If no DESFire card can be found within a number of seconds or another problem has arisen preventing a configuration to be written on the DESFire card the first error message will be shown, if the card has been used earlier then the application has already been written and the second error message will be shown.

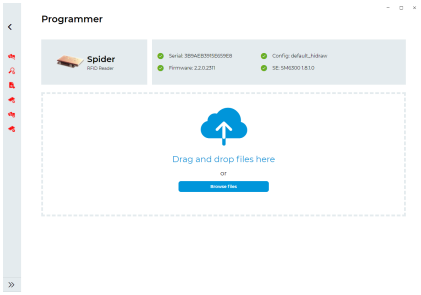
If no error messages occur, you should see that message that the process has finished successfully.



The configuration card can now be used to write it's configuration file onto other Spider RFID readers.

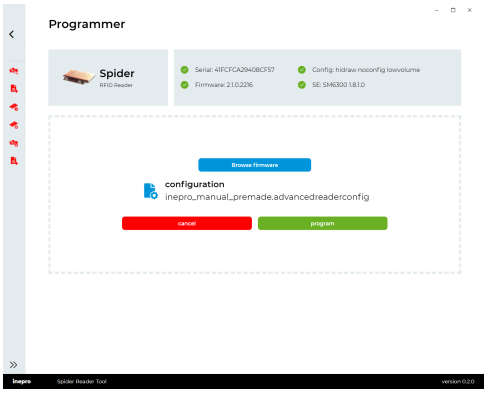
Programmer

Load configuration or firmware files onto Spider RFID Reader to change the configuration or update the firmware.

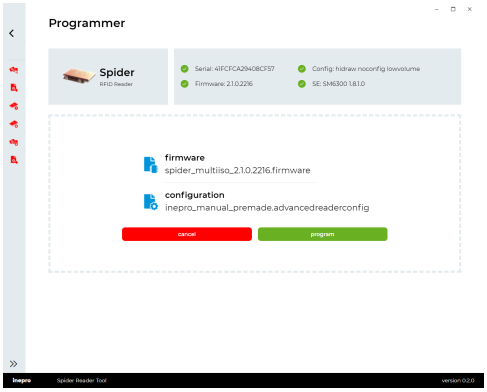


The top bar has the Spider RFID Reader type and status.

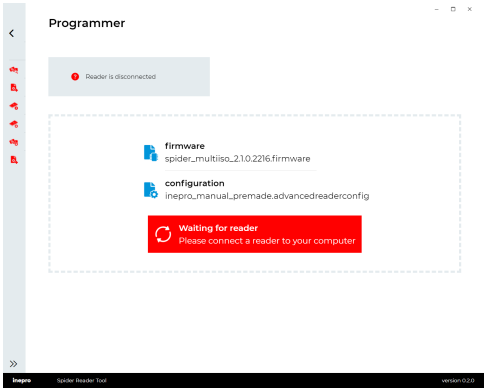
1. Drag and drop a *.readerconfig or *.advancedreaderconfig (this is a Spider Settings file) and/or a *.firmware file into the drag-and-drop-area or click the “Browse files” button.



2. After selecting a configuration file you can either select a firmware file in addition to the configuration file (in the same manner as the configuration file was selected) of continue just using the configuration file by clicking “program”. The order in which the configuration file and the firmware file are added does not matter.

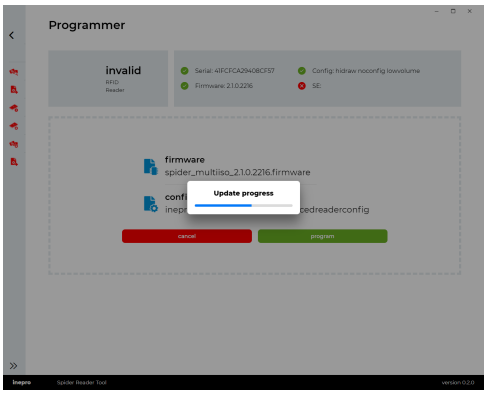


In the screen shot above both a configuration file as a firmware file have been selected to program onto the Spider RFID reader.

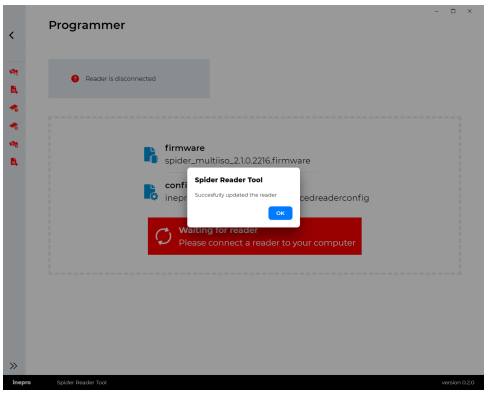


⚠ If the Spider RFID Reader is disconnected, it will no longer be possible to program it. An error message will be shown if the Spider RFID Reader has been disconnected. Please make sure the Spider RFID Reader stays connected during this process.

3. Click 'Program'



Programming in progress



4. When programming the Spider RFID Reader has finished this message will appear, please confirm this message ■

Appendix A: All other config settings

In this appendix we will try to explain all the settings that can be edited with this tool that have not been treated earlier in this document.

ID Formatting

The card ID can be formatted by using the settings in this table.

ID Formatting		
Name	**Default Value**	**Description**
USB Backward Compatible Output	Full output	Select whether the classic or full output is used **Full output** | Classic mode using upper case hexadecimal (card ID only) | Classic mode using upper case hexadecimal (card ID only, supported from version 43.1.1194) | Full output format using classic output mode instead of 32-bit ID (supported from version 51.0.1442)
USB Backwards Compatible Output Byte Order	default	Set the byte order of the old output format when using USB HID interface (compatible with Ricoh readers) **default** | reversed
USB Backward Compatible Output Decimal	hexadecimal	Configures the old output format for USB HID interface (compatible with Ricoh readers) **hexadecimal** | decimal | 16-bit byte swapped decimal output (UID is byte swapped and right-shifted with one bit, then the decimal representation of bits 0-15 is used as output. (valid since version 41.2)
USB Backward Compatible Output ASCII Prefix	<empty>	Define an optional prefix which is output in front of the card number. This setting only works in backward compatible / classic mode <string value>
USB Backwards Compatible Output Length	0	Set the length of the old output format. Use '0' for automatic and '255' to string leading zeroes.. This setting affects the resulting output including binary UIDs and ASCII/BCD numbers read from transponder data **0**-255
USB Backward Compatible Output Offset	0	Set the offset of the old output format. This setting affects the resulting output including binary UIDs and ASCII/BCD numbers read from transponder data **0** | <numeric>

Appendix A: All other config settings

ID Formatting		
Name	**Default Value**	**Description**
USB Backward Compatible UID Offset	0	Set the offset of the UID in old output format mode. . This setting only works on (binary) UIDs, not on ASCII/BCD numbers which are read from transponder data **0** | <numeric>
USB Backward Compatible UID Length	0	Set the size of the UID in old output format mode. Use '0' for full size. This setting only works on (binary) UIDs, not on ASCII/BCD numbers which are read from transponder data **0** | <numeric>
USB Backward Compatible UID Format	default	Define the output formatting of the UID in old format mode. This setting only works on (binary) UIDs, not on ASCII/BCD numbers which are read from transponder data **default** | HEX 10 | ZK (DEZ 20) | IK2 (DEZ 14) | IK3 (DEZ 15) | DEZ 10 | DEZ 8 | DEZ 5.5 | DEZ 3.5A | DEZ 3.5B | DEZ 3.5C
USB Backward Compatible 32-bit UID	disabled	Define the output formatting of the UID in old format mode **disabled** | enabled
USB Backward Compatible SHA256	disabled	Convert the reader output to an SHA256 hash of 32 bytes. This setting only works in backward compatible / classic mode **disabled** | enabled
Prepend Facility Code	disabled	Prepend the facility code to the reader output **disabled** | use ':' as separator character (FC:UID) | Do not use separator character (FCUID)
USB Backward Compatible ASCII Hex	disabled	Convert the reader output to "ASCII hexadecimal" (example: 345EF => 3334354546). This setting only works in backward compatible / classic mode **disabled** | enabled

Appendix A: All other config settings

General settings

General settings		
Name	**Default Value**	**Description**
USB Device Interface	HID	Defins the Device interface for USB. By default this is the **H**uman **I**nterface **D**evice (HID, not to be confused with the card technology **H**ughes **I**dentification **D**evices (HID®). **HID** | CDC ACM (virtual COM port) | RAW HID (COM port functionality without driver) | CDC ETH (ethernet adapter) | CDC ACM device with asynchronous card data output (HID mode functionality on virtual COM port interface) | RS232 device with asynchronous card data output (HID mode functionality on serial port interface)
USB Ignore Card Removed	Generate output when card is removed	Configure whether or not to generate output when card is removed from reader **Generate output when card is removed** | ignore when card is removed
USB HID Enter Key	keypad enter (0x58)	Set which USB HID enter key is used to terminate the ID output string **keypad enter (0x58)** | enter (0x28)
Beeper Volume	low	Set the volume level of the beeper off | **low** | medium | high
LED Behaviour	enabled	Define how the LEDs should behave disabled | **enabled**
Technology Enabling Mode	All technologies are disabled by default, unless the configuration is empty. In this case, all non-conflicting technologies will be enabled behaving like in demo mode	Configure the way in which transponder technologies are enabled/disabled **All technologies are disabled by default, unless the configuration is empty. In this case, all non-conflicting technologies will be enabled behaving like in demo mode** | All non-conflicting technologies will be enabled by default unless they are explicitly disabled in the configuration
Config Card Acceptance Time	always	Define a time interval during which configuration cards are accepted. Value is defined in seconds since boot. never | 30 | 60 | 120 | 240 | **always**

Appendix A: All other config settings

General settings		
Name	**Default Value**	**Description**
Avoid Numerical Output	disabled	Avoid using numerical characters in reader output. Characters '0...9' are replaced by 'g...p'. When this mode is active, a '#' is prepended to the reader output **disabled** | enabled
USB HID Report Interval	500	Set the maximum amount of characters to be sent over USB HID in characters per second **500** | 250 | 125 | 62 | 31 | 15
KUARIO Output Format	disabled	Convert the reader output to a KUARIO compatible encrypted format **disabled** | enabled
Encryption Key	00000000000000000000000000000000	Encrypt all HID keyboard data with the configured AES key **00000000000000000000000000000000** | <16- or 24 byte value>
Technology Scan Interval	255	Set the delay (in milliseconds) between scanning different 13.56MHz technologies. The default is "255", where this sets a 15ms delay in default config mode and 0ms delay when one or more technologies are explicitly enabled 0-**255**
Test Mode	test mode disabled; normal reader operation	Enable one of the test modes **Test mode disabled; normal reader operation** | Test mode 1; keep 13.56MHz fields enabled continuously | Test mode 2; keep 125kHz fields enabled continuously
Remote Wake Up	disabled	Wake up the host when a card is detected **disabled** | enabled

Appendix A: All other config settings

Legic settings

LEGIC RFID chips use 13,56 MHz frequency and are one of the most frequently applied RFID chip cards. Accordingly, LEGIC chip cards might be used for employee ID cards, membership cards, season tickets and eTickets.

LEGIC Chip Card Overview

Chip	Frequency	Memory	ISO
LEGIC prime MIM 256 Chip Card	13,56 MHz	256 Byte	-
LEGIC prime MIM 1024	13,56 MHz	1024 Byte	-
LEGIC advant ATC 1024	13,56 MHz	1024 Bit	15693
LEGIC advant ATC 2048-MP	13,56 MHz	2048 Byte	14443A
LEGIC advant ATC 4096-MP	13,56 MHz	4096 Byte	14443A
LEGIC CTC4096 MM410	13,56 MHz	4096 Byte	14443A, 15693

Legic settings		
Name	**Default Value**	**Description**
LEGIC Advant Support	disabled	Defines whether or not to enable Legic Advant transponders **disabled** | enabled
LEGIC Prime Support	disabled	Defines whether or not to enable Legic Prime transponders **disabled** | enabled
LEGIC Mifare Support	disabled	Defines whether or not Mifare support is required for Legic Advant reader **disabled** | enabled
LEGIC UID Byte Order	default	UID byte order of Legic cards. Affects the 32-bit card ID, not the raw binary card data **default** | Reversed byte order | config.item.caption.prime-capability | config.item.caption.advant-reversed-prime-capability
LEGIC ID Segement Mode	none	Type of Legic ID segment to search for **none** | KGH | Interflex | TimeLink | packed BCD | ASCII | Binary BE | Binary LE | ASCII Binary | Packed PCD nibbles | BER | KGH with one leading zero | Legic Access segment
LEGIC Segment Search Stamp	00	The Legic ID segment stamp to look for. To be used with setting Legic Segment Mode. **00** | <numeric>

Appendix A: All other config settings

Legic settings		
Name	**Default Value**	**Description**
LEGIC Read Offset	0	When the setting "Legic ID Segment Mode" is set to either ASCII, BCD or Binary mode, the number to be read starts from the offset specified in this setting.For Prime transponders, the read offset from in this setting starts after the length of setting "Legic Segment Search Stamp". For Legic Advant transponders, the read offset of this setting starts after the stamp length **0**-255
LEGIC Read Length	0	When the value of this setting is greater than zero and the setting "Legic ID Segment Mode" is set to either ASCII, BCD or Binary mode, the number is read with the specified length of this setting **0**-255
LEGIC CRC Offset	0	When setting "Legic ID Segment Mode" is set to either ASCII, BCD or Binary mode and the value of this setting is non-zero, a CRC will be calculated and checked from the specified offset in this setting and the length defined in "Legic Read CRC Length **0**-255
LEGIC CRC Length	0	When this setting is non-zero and the setting "Legic ID Segment Mode" is set to either ASCII, BCD or Binary mode, a crc will be calculated and checked from the CRC offset defined in "Legic Read CRC Offset" over the specified length. **0**-255
LEGIC CRC Flags	8 Bit CRC: Do not include the stamp in calculation	When setting "LEGIC ID Segment Mode" is set to one of the ASCII/BCD/BINARY modes and setting "LEGIC Read CRC Length" is nonzero, a CRC will be calculated and checked. **8 Bit CRC: Do not include the stamp in calculation** | 8 Bit CRC: Include the stamp in calculation | 16 Bit CRC: Do not include the stamp in calculation | 16 Bit CRC: Include the stamp in calculation

Appendix A: All other config settings

Legic settings		
Name	**Default Value**	**Description**
LEGIC UID Fallback	disabled	When the setting "Legic ID Segment Mode" is non-zero and the configured ID segment is not found, a fallback to the Legic UID can be done. In case the transponder is not positioned correctly, it is possible that the UID is used regardless of the segment ID availability. **disabled** | enabled | Legic prime transponders only | Legic Advant transponders only
LEGIC Advant Standards	ISO14443A + ISO15693	Define which Legic Advant standards should be supported. When this setting is absent, all standards are supported by default. ISO14443A | ISO15693 | **ISO14443A + ISO15693**
LEGIC Prepend Stamp Bytes To Segment ID	0	Define the amount of bytes from the ID sector stamp to be prepended to the user ID. Requires the setting "Legic ID segment mode" to be other than "none" **0** | <numeric>
LEGIC Prepend Zeroes To Segment ID	0	Define the amount of zeroes to prepend to the user ID when both the value of this setting and "Legic ID Segment Mode" are non-zero **0**-255

HID® iCLASS®

HID manufactures and licenses several types of technologies, from Wiegand products to 13.56 MHz iCLASS®, MIFARE, and DESFire, as well as the 125 kHz Indala® and Prox cards. Migration readers from various 125 kHz Prox technologies to 13.56 MHz iCLASS® were introduced in 2007.

HID® iCLASS®		
Name	**Default Value**	**Description**
HID® iCLASS® Support	disabled	Defines whether or not to enable HID® iCLASS® tags **disabled** | enabled
HID® iCLASS® Seos® Support	disabled	Defines whether or not to enable HID® iCLASS® Seos® tags **disabled** | enabled

Appendix A: All other config settings

HID® iCLASS®		
Name	**Default Value**	**Description**
HID® iCLASS® Format	CSN	Define the HID® iCLASS® access control ID format. In case of CSN (240), the HID® iCLASS® transponder serial number is used instead of the access control ID data. automatic | H10301 | H10302 | H10304 | H10306 | C1000-35 | P10004 | ARAS36 | G10901 | C1000-48 | NFP | NFP1 | PAL | PAH | BDC | WFH | WBC | Simplex36 | RBH50 | HID137 | **CSN** | RAW without start and end bit | RAW without start bit | RAW
HID® iCLASS® Facillity Code	0	Define the HID® iCLASS® transponder facility code. **0** | <numeric>

Mifare® Classic

MIFARE is the NXP Semiconductors-owned trademark of a series of integrated circuit (IC) chips used in contactless smart cards and proximity cards.

The brand name covers proprietary solutions based upon various levels of the ISO/IEC 14443 Type A 13.56 MHz contactless smart card standard. It uses AES and DES/Triple-DES encryption standards, as well as an older proprietary encryption algorithm, Crypto-1.

These Mifare settings are for Mifare® Classic which employs a proprietary protocol compliant to parts 1–3 of ISO/IEC 14443 Type A, with an NXP proprietary security protocol for authentication and ciphering. Subtype: Mifare® Classic EV1 (other subtypes are no longer in use).

Mifare® Classic		
Name	**Default Value**	**Description**
Mifare® Classic Support	disabled	Defines whether or not to enable Mifare® Classic tags **disabled** | enabled
Mifare ID Mode	ISO (UID)	Type of Mifare ID to Use **ISO (UID)** | Edeka sector
Mifare ID Sector	1	Mifare sector to read from 0, **1**-15
Mifare ID Sector Block	0	Mifare block (within sector) to read the ID from **0**-2
Mifare ID Offset	0	Offset (within block) to read the Mifare ID from **0**-15

Appendix A: All other config settings

Mifare® Classic		
Name	**Default Value**	**Description**
Mifare ID Length	0	Length of Mifare ID **0**-16
Mifare ID AID	0000	AID of Mifare ID sector **0000** | [4 digit number]
Mifare ID Key Type	A	Type of Mifare key **A** | B
Mifare ID Key	000000000000	Key of Mifare ID sector **000000000000** | [12-digit number]

Mifare DESFire®

Mifare DESFire® are contactless ICs that comply with parts 3 and 4 of ISO/IEC 14443-4 Type A with a mask-ROM operating system from NXP. The DES in the name refers to the use of a **D**ata **E**ncryption **S**tandard, two-key 3DES, three-key 3DES and AES encryption; while Fire is an acronym for Fast, innovative, reliable, and enhanced. Subtypes: Mifare DESFire® EV1, Mifare DESFire® EV2, Mifare DESFire® EV3.

DESFire settings		
Name	**Default Value**	**Description**
DESfire Support	disabled	Defines whether or not to enable DESFire tags **disabled** | enabled
DESFire Authentication Mode	Do not check; use UID	Defines how a presented DESFire card has to be handled **Do not check; use UID** | Check version key (indentifies valid DESFire card) and use UID | check inepro PICC and use UID | check inepro application and use UID | Read and use file data | Check applicatio and use UID
DESFire Authentication Key Type	DES	Defines the type of key used for DESFire authentication **DES** | AES | ISO
DESFire Authentication Key Number	0	Key number used for DESFire authentication **0** | <numeric>
DESFire Authentication Key	00000000000000000000000000000000	DESFire authentication key **00000000000000000000000000000000** | <16- or 24 byte value>
DESFire Authentication AID	000000	DESFire application ID used for authentication **000000** | [6 digit number]
DESFire Authentication File Number	0	File number used for DESFire authentication **0** | <numeric>

Appendix A: All other config settings

DESFire settings		
Name	**Default Value**	**Description**
DESFire Authentication File Offset	0	Offset of data in the file used for DESFire authentication **0** | <numeric>
DESFire Authentication File Size	0	Defining the size of the data in the file used for DESFire authentication **0** | <numeric>
DESFire Authentication File Format	ASCII	Define the format of the data in the DESFire file, which is used for DESFire authentication **ASCII** | BCD | packed BCD | Binary BE | Binary LE | ASCII decimal to binary | packed BCD measured in nibbles
DESFire Authentication File Plain	encrypted	Expect the DESFire file to contain either encrypted or plain data **encrypted** | plain

Mifare® Plus

MIFARE® Plus
Drop-in replacement for Mifare® Classic with certified security level (AES-128 based) and is fully backward compatible with Mifare® Classic. Subtypes MIFARE® Plus S, MIFARE® Plus X and MIFARE® Plus SE.

MifarePlus settings		
Name	**Default Value**	**Description**
Mifare® Plus Support	disabled	Defines whether or not to enable Mifare Plus tags **disabled** | enabled
Mifare® Plus ID Mode	ISO (UID)	Type of Mifare ID to use **ISO (UID)** | Packed BCD | ASCII | Binary BE | Binary LE
Mifare® Plus ID Sector	0	Mifare sector to read the ID from **0**-23
Mifare® Plus ID Sector Block	0	Mifare block (within sector) to read the ID from **0**-2
Mifare® Plus ID Offset	0	Offset (within block) to read the Mifare ID from **0**-15
Mifare® Plus ID Length	1	Length of Mifare ID **1**-16
Mifare® Plus ID Key Type	A	Type of Mifare key **A** | B

Appendix A: All other config settings

MifarePlus settings		
Name	**Default Value**	**Description**
Mifare® Plus ID Key	000000000000	Key of Mifare ID sector **000000000000** | [16-byte value]
Mifare® Plus ID Plain Transfer	encrypted	Set whether to use encrypted or plain data transfer **encrypted**

Other Mifare® family

Mifare® Classic, DESfire®, Plus have been treated, this section is about the other Mifare® family members.

Other Mifare® family		
Name	**Default Value**	**Description**
Mifare® Ultralight Support	disabled	Defines whether or not to enable Mifare Ultralight tags **disabled** | enabled
Mifare® Mini Support	disabled	Defines whether or not to enable Mifare Mini tags **disabled** | enabled

Sony FeliCa settings

FeliCa is a contactless RFID smart card system from Sony in Japan, primarily used in electronic money cards. The name stands for Felicity Card. FeliCa's encryption key is dynamically generated each time mutual authentication is performed, preventing fraud such as impersonation.

FeliCa complies with JIS: X6319-4: Specification of implementation for integrated circuit(s) cards - Part 4: High speed proximity cards. The standard is regulated by JICSAP (Japan IC Card System Application Council).

Sony FeliCa settings		
Name	**Default Value**	**Description**
FeliCa Support	disabled	Defines whether or not to enable FeliCa tags **disabled** | enabled
FeliCa ID Mode	ISO (UID)	Type of FeliCa ID to use **ISO (UID)** | Packed BCD | ASCII | Binary BE | Binary LE | ASCII Binary | Packed BCD measured in nibbles
FeliCa ID System	0000	FeliCa System to read the ID from **0000** | [4 digit number]

Appendix A: All other config settings

Sony FeliCa settings		
Name	**Default Value**	**Description**
FeliCa ID Service	0	FeliCa Service (within a System) to read the ID from **0** | <numeric>
FeliCa ID Block	0	FeliCa Block (within a Service) to read the ID from **0** | <numeric>
FeliCa ID Offset	0	Read the FeliCa ID from within a block with this offset **0**-15
FeliCa ID Length	0	Length of the ID to be read **0**-16

Appendix A: All other config settings

Other HF Settings

ISO/IEC 15693
ISO/IEC 15693, is an ISO/IEC standard for vicinity cards, i.e. cards which can be read from a greater distance as compared with proximity cards. Such cards can normally be read out by a reader without being powered themselves, as the reader will supply the necessary power to the card over the air (wireless).

ISO/IEC 15693 systems operate at the 13.56 MHz frequency. As the vicinity cards have to operate at a greater distance, the necessary magnetic field is less (0.15 to 5 A/m) than that for a proximity card (1.5 to 7.5 A/m).

ISO14443A and ISO14443B
ISO/IEC 14443 Identification cards -- Contactless integrated circuit cards -- Proximity cards is an international standard that defines proximity cards used for identification, and the transmission protocols for communicating with it.

Cards may be Type A and Type B, both of which communicate via radio at 13.56 MHz (RFID HF). The main differences between these types concern modulation methods, coding schemes and protocol initialization procedures.

INSIDE
INSIDE Secure card technology is used in Visa Token Service (VTS) and Mastercard Digital Enablement Service (MDES) as well as other organizations that require stringent payment-related security.

NTAG
NTAG is used for **N**ear **F**ield **C**ommunication (NFC). In general, NTAG® Series ICs are compatible with every NFC mobile device. The Operative Frequency of NFC Tags is 13.56 MHz.

Other HF Settings settings		
Name	**Default Value**	**Description**
Other ISO15693 Support	disabled	Defines whether or not to enable other/unknown ISO15693 tags (in UID mode) **disabled** | enabled
Other ISO14443A Support	disabled	Defines whether or not to enable other/unknown ISO14443A tags (in UID mode) **disabled** | enabled
Other ISO14443B Support	disabled	Defines whether or not to enable other/unknown ISO14443B tags (in UID mode) **disabled** | enabled
INSIDE Support	disabled	Defines whether or not to enable INSIDE tags **disabled** | enabled
NTAG Support	disabled	Defines whether or not to enable NTAG transponders **disabled** | enabled

Appendix A: All other config settings

Other HF Settings settings		
Name	**Default Value**	**Description**
UID Byte Order	default	Byte order of raw UID data. Applies to four or seven byte UIDs of ISO14443A transponders. Affects both 32-bit card ID and the raw binary card data. Does not affect ISO14443A Legic Advant and ISO15693 transponders **default** | reversed byte order

Hitag®

HITAG® products support the low-frequency (LF) RFID market—delivering high reliability, robust performance and safe data transmission. HITAG® transponder ICs operate at a frequency range of 125 kHz and offer the advantage of operating in harsh environments. Transponder ICs in the HITAG® family are compliant with ISO 11784/85 and ISO 14223 animal identification standards as well as the industrial ISO 18000-2. HITAG® transponder IC's feature an ultra-low-power design to support the long read ranges like those needed for livestock tracking.

Hitag®		
Name	**Default Value**	**Description**
Hitag1 Support	disabled	Defines whether or not to enable Hitag1 tags **disabled** | enabled
Hitag2 Support	disabled	Defines whether or not to enable Hitag2 tags **disabled** | enabled
HitagS Support	disabled	Defines whether or not to enable HitagS tags **disabled** | enabled
Hitag1 Page Address	0	Defines which HITAG1/HITAGS page should be used instead of the HITAG UID. **0** - 63

HID® Prox and Indala®

HID® Prox

HID Proximity credentials offer an affordable yet robust solution for entry-level access control. It operates on a 125 kHz frequention.

Indala®

Indala® proximity cards from HID Global are 125 kHz credentials and feature an added layer of access control security.

HID Prox and Indala®		
Name	**Default Value**	**Description**
HID Prox Support	disabled	Defines whether or not to enable HID Prox tags **disabled** | enabled

Appendix A: All other config settings

HID Prox and Indala®		
Name	**Default Value**	**Description**
Indala® Support	disabled	Defines whether or not to enable Indala® tags **disabled** | enabled
Indala®224 Support	disabled	Defines whether or not to enable Indala® 224-bit transponders **disabled** | enabled
Wiegand Format	automatic	Defines the Wiegand bit stream format for 32-bit card IDs and raw binary card data **automatic** | H10301 | H10302 | H10304 | H10306 | C1000 | P10004 | ARAS36 | G10901 | NFP | NFP1 | PAL | PAH | BDC | WFH | WBC | RAW without start and end bit | RAW without start bit | RAW
Wiegand Facility Code	0	Defines the facility code **0** | <numeric>
Wiegand Tag Type	HID Prox reader	Defines the wiegand tag type **HID Prox reader** | Indala® reader
Indala® Format	automatic	Set the Wiegand bit stream for Indala® transponders. Affects both 32-bit card ID and raw binary card data **automatic** | H10301 | H10302 | H10304 | H10306 | C1000 | P10004 | ARAS36 | G10901 | NFP | NFP1 | PAL | PAH | BDC | WFH | WBC | RAW without start and end bit | RAW without start bit | RAW
Indala® Facility Code	0	Define the facility code for Indala® transponders **0** | <numeric>

Other LF settings

UNIQUE
Unique 125 kHz is an RFID standard for transmitting/receiving data via radio waves at 125 kHz carrier frequency. Data exchange does not require optical visibility of the proximity card and the proximity reader, required by the barcode systems.
Unique 125 kHz standard features 64-bit ROM (Read-Only Memory). The proximity cards are factory programmed. In theory, 64 bits allows to write 264 = 18 446 744 073 709 600 000 unique numbers.

Q5
The Q5 is a 'smart' tag, capable of emulating other standard tags, including the ID number. 125kHz RFID IC, Read/Write 264 bits.

Appendix A: All other config settings

EM4X05
The EM Microelectronic EM4x05 range implement the animal tagging standard ISO-11784 'Radio-frequency identification of
animals - Code structure' and ISO-11785 'Radio-frequency identification of animals - Technical concept' (also known as FDX-B). These chips are ID-only transponders, operating at 134.2kHz and storing 128 bits of data, 64 bits of which are the ID.

EM4X50
The EM Microelectronic EM4x50 (read and write) tag contains a unique read-only serial number (one of 2^32, or 4,294,967,296 possible combinations) and 116 bytes of user data area stored in a non-volatile EEPROM (Electrically Erasable Read Only Memory). Also used for animal tagging.

TI-RFID
Texas Instruments Radio Frequency Identification (TI-RFid™) Systems is an industry leader in radio frequency identification (RFID) technology and the world's largest integrated manufacturer of RFID tags, RFID smart labels, and RFID reader systems. Approaching 500 million RFID tags manufactured, TI-RFid™ technology is used in a broad range of RFID applications worldwide including automotive, contactless payments, laundry, library, livestock, pharmaceutical & healthcare, retail supply chain management, and ticketing.

NEDAP
Nedap was founded in 1929, originally abbreviated as Nafa, and changed to its current name in 1933. The company was originally located in Amsterdam but moved to Groenlo in 1946. In 1947, Nedap took the step of going public, and in 1949 the listing on the Amsterdam stock exchange became definite.
In 1999, Nedap started the new market group Nedap Healthcare. It began with a small, handy RFID reader for the healthcare provider.

IOProx (XSF)
ioProx XSF This is a popular access control format which offers a higher level of security compared to most access control formats.

Pyramid
Pyramid Series Proximity® is recognized as an electronic-security benchmark for 125-kHz OEM proximity readers and credentials. Pyramid credentials, proximity cards and tags, are passive devices, eliminate maintenance by requiring no battery.

AWID
AWID combines the latest technology with design expertise for radio frequency (RF), antenna and communication systems that offer professional engineering services in access control systems.

TimeTouch
TimeTouch is also an access control system company.

ProxLite
Format of the Casi/ProxLite Readers.

Keri / Keri NXT
Keri manufactures its own proximity access key cards and tags, all available in Keri format, NXT format.

Appendix A: All other config settings

Cotag
Cotag is a very convenient and user-friendly card technology that is unique on the market. When Cotag technology is used, it is possible to have both proximity and hands-free card reading in the same system. All Cotag readers can read both active (long-range) and passive (proximity) Cotag cards and tags, and both types of cards and tags can be mixed in the same system to provide ultimate convenience and cost-efficiency. This is particularly useful in cases where some cardholders are disabled or where hands are often occupied, such as in a hospital or a warehouse. Cotag is more secure than standard 125 kHz technology because the content on the card is protected and there is no equipment available outside Siemens to copy or tamper with the cards.

TK5561
The TK5561A-PP is a complete transponder integrating all important functions for immobilizer and identification systems. It consists of a plastic cube which accommodates the crypto IDIC e5561A and the antenna realized as tuned LC-circuit. The TK5561A-PP is a R/W crypto transponder for applications which demand higher security levels than those which standard R/W transponders can fulfil. For this reason, the TK5561A-PP has an additional encryption algorithm block which enables a base station to authenticate the transponder. Any attempt to fake the base station with a wrong transponder will be recognized immediately. For authentication, the base station transmits a challenge to the TK5561A-PP. This challenge is encrypted by both IC and base station. Both should possess the same secret key. Only then the result be expected to be equal.

Gallagher/Cardax
Cardax primarily supports its own proprietary format, using Amplitude Shift Keying (ASK) modulation. When connected in the Cardax FT system, the readers are also capable of reading the other selected third party formats using Phase Shift Keying (PSK) and Frequency Shift Keying (FSK).

FDX-B
The FDXB technology is ideally suited for harsh environments where electrical noise, moisture, or large metal surfaces are present. Often used to 'chip' (inject an ID tag) pet animals.

PAC/Stanley
Simplicity at its finest. These handy RFID tags are highly reliable proximity tags. These tags use PAC 153KHz proximity technology. Simply attach to a keyring and use as needed. Because of the batteryless design, these RFID tags can be used again and again.

NexWatch
When NexWatch invented proximity access control technology in 1972 it revolutionized the security industry. Because of its convenience, dependability, enhanced security, low maintenance, and high end-user acceptance proximity has become the most desired access control technology available. NexWatch's latest innovations in access control technology include: multiple-technology ISO-compliant cards; low-cost proximity badges, and; long-range proximity cards. Each card provides the convenience of hands-free access, while acting as a key to a secure and dependable access control system.

Designed for use with the award-winning DigiReader Series Digital Proximity Readers, NexWatch's digital cards provide some of the longest read ranges available for passive proximity cards. The cards are powered by the reader; they require neither batteries nor maintenance. Sturdy, world-class construction, advanced technology and unique numbers for each card make them virtually impossible to counterfeit.

Appendix A: All other config settings

Deister Electronic
A specifically developed data format, “deister smartFrame”, ensures reliable, tamper-proof identification of Deister Electronic transponders.

G-Prox II
The G-Prox™ II proximity access cards permit convenient contact-less operation without the wear of a physical contact card. They are available blank (un-programmed) or in either industry standard 26 bit format or the VEREX exclusive 36 bit format. All G-Prox™ II cards and keychain tokens are fully compatible with all of the G-Prox™ II proximity reader line. User Programmable. The optional secure Lock Codes enable the installer to lock cards and readers to a specific and unique code providing enhanced security and anti-takeover features. Avaliable as G-ProxCard, G-ProxPhoto, G-ProxTwin and G-ProxKey access control proximity cards compatible with G-Prox II readers.

NeocardPROX
The NEOCard Prox is a reprogrammable passive proximity tag, 125 KHz. The tag is of the key ring type, it's reprogrammable (read/write capabilities) and serialised with laser marked number.
Dimensions: 37 x 49 x 10 mm. Reprogrammable passive proximity card, 125 KHz. Adheres to the ISO format.

Other LF settings		
Name	**Default Value**	**Description**
UNIQUE Support	disabled	Defines whether or not to enable UNIQUE tags **disabled** | enabled
Q5 Support	disabled	Defines whether or not to enable Q5 tags **disabled** | enabled
EM4X05 Support	disabled	Defines whether or not to enable EM4X05 tags **disabled** | enabled
EM4X50 Support	disabled	Defines whether or not to enable EM4X50 tags **disabled** | enabled
TI-RFID Support	disabled	Defines whether or not to enable TI-RFID tags **disabled** | enabled
NEDAP Support	disabled	Defines whether or not to enable NEDAP 120kHz transponders (128-bit raw bit stream) **disabled** | enabled
IOProx (XSF) Support	disabled	Defines whether or not to enable IOProx (XSF) tags **disabled** | enabled
Pyramid Support	disabled	Defines whether or not to enable Pyramid (Farpointe Data) transponders **disabled** | enabled
AWID Support	disabled	Defines whether or not to enable AWID transponders **disabled** | enabled
TimeTouch Support	disabled	Defines whether or not to enable TimeTouch transponders **disabled** | enabled

Appendix A: All other config settings

Other LF settings		
Name	**Default Value**	**Description**
ProxLite Support	disabled	Defines whether or not to enable ProxLite transponders **disabled** | enabled
Keri NXT Support	disabled	Defines whether or not to enable Keri NXT transponders **disabled** | enabled
Cotag Support	disabled	Defines whether or not to enable Cotag transponders **disabled** | enabled
TK5561 Support	disabled	Defines whether or not to enable TK5561 transponders **disabled** | enabled
Gallagher/Cardax Support	disabled	Defines whether or not to enable Gallagher/Cardax transponders **disabled** | enabled
FDX-B Support	disabled	Defines whether or not to enable FDX-B transponders **disabled** | enabled
PAC/Stanley Support	disabled	Defines whether or not to enable PAC/Stanley transponders **disabled** | enabled
NexWatch Support	disabled	Defines whether or not to enable NexWatch transponders **disabled** | enabled
NeocardPROX Support	disabled	Defines whether or not to enable NeocardPROX transponders **disabled** | enabled
Deister Electronic Support	disabled	Defines whether or not to enable Deister Electronic Support transponders **disabled** | enabled
Keri Support	disabled	Defines whether or not to enable Keri transponders **disabled** | enabled
Keri Facility Code	0	Defines the facility code for Keri transponders. Use value 0 to disable. **0** | <numeric>
G-Prox II Support	disabled	Defines whether or not to enable G-Prox II transponders **disabled** | enabled
G-Prox II Facility Code	0	Defines the facility code for G-Prox II transponders. Use value 0 to disable **0** | <numeric>

Appendix A: All other config settings

Other LF settings		
Name	**Default Value**	**Description**
UNIQUE Inverted	disabled	Defines whether or not to invert the UID of UNIQUE tags **disabled** | enabled
EM4X05 Inverted	disabled	Defines whether or not to invert the UID of EM4X05 tags **disabled** | enabled
IOProx Facility Code	0	Define the IOProx (XSF) transponder facility code **0** | <numeric>
Pyramid Facility Code	0	Define the Pyramid (Farpointe Data) transponder facility code **0** | <numeric>
AWID Facility Code	0	Define the AWID transponder facility code **0** | <numeric>
AWID Format	automatic	Set the wiegand bit stream format for AWID transponders. Affects both 32-bit card ID and raw binary card data **automatic** | H10301 | H10302 | H10304 | H10306 | C1000 | P10004 | ARAS36 | G10901 | NFP | NFP1 | PAL | PAH | BDC | WFH | WBC | RAW without start and end bit | RAW without start bit | RAW
NeocardPROX Page	default page	Defines which NeocardPROX page the reader should use **default page** | alternate page
Gain and Filter	8	gain0/gain1/filterL/filterH 1-**8**-16

Appendix A: All other config settings

BLE (BlueTooth Low Energy) and NFC

Bluetooth Low Energy (Blutooth Smart)
Bluetooth Low Energy (Bluetooth LE, colloquially BLE, formerly marketed as Bluetooth Smart) is a wireless personal area network technology designed and marketed by the Bluetooth Special Interest Group (Bluetooth SIG).
Compared to Classic Bluetooth, Bluetooth Low Energy is intended to provide considerably reduced power consumption and cost while maintaining a similar communication range. Mobile operating systems including iOS, Android, Windows Phone and BlackBerry, as well as macOS, Linux, Windows 8 and Windows 10, natively support Bluetooth Low Energy.

NFC
Near **F**ield **C**ommunication (NFC) is a set of communication protocols that enables communication between two electronic devices over a distance of 4 cm (11⁄2 in) or less. NFC offers a low-speed connection through a simple setup that can be used to bootstrap more-capable wireless connections. Like other "proximity card" technologies, NFC is based on inductive coupling between two so-called antennas present on NFC-enabled devices—for example a smartphone and a printer—communicating in one or both directions, using a frequency of 13.56 MHz in the globally available unlicensed radio frequency ISM band using the ISO/IEC 18000-3 air interface standard at data rates ranging from 106 to 424 kbit/s. FC standards cover communications protocols and data exchange formats and are based on existing radio-frequency identification (RFID) standards including ISO/IEC 14443 and FeliCa.

BLE (BlueTooth Low Energy) and NFC		
Name	**Default Value**	**Description**
BLE Support	disabled	Defines whether or not to accept Spider ID app connections. **disabled** | enabled
NFC Support	disabled	Defines whether or not to enable NFC tags **disabled** | enabled

Appendix A: All other config settings

<table>
<tr><th colspan="3">BLE (BlueTooth Low Energy) and NFC</th></tr>
<tr><th>Name</th><th>Default Value</th><th>Description</th></tr>
<tr><td>BLE RSSI Theshold Mode</td><td>disabled</td><td>Defines how the BLE RSSI threshold value should be used.

These values determine when a signal (of a certain strength) is seen as a valid signal. By default the fixed RSSI threshold values of the Spider Apps are used. The RSSI threshold for the Spider ID App is -60 dBm. In meters, this is about 1 meter distance between the Spider RFID Reader and the phone. If disabled, use the threshold value specified by Spider Apps. It is possible to assign the Spider RFID Reader it's own RSSI Threshold level. This setting defines how the RSSI threshold will be calculated between the Spider RFID Reader and the phone.

disabled | RSSI threshold config settings; Ignore SpiderApp threshold setting | Highest value of RSSI threshold setting and SpiderApp threshold setting | Lowest value of RSSI threshold setting and SpiderApp threshold setting | Sum of the RSSI threshold setting and SpiderApp threshold setting</td></tr>
<tr><td>BLE RSSI Threshold Value</td><td>0</td><td>Defines the BLE RSSI threshold value in dBm.

Signal Strength is measured in dBm and cannot be converted to distance directly. The distances given are by approximation as that distance is depended on circumstances like wall material, phone model and other variables. As a such they might drastically differ from the values in the environment situation in which your reader operates.

Some practical values are:
<table>
<tr><th>RSSI in dBm</th><th>Distance in meters (approx.)</th></tr>
<tr><td>-50</td><td>0.5m</td></tr>
<tr><td>-60</td><td>1m</td></tr>
<tr><td>-70</td><td>2m</td></tr>
<tr><td>-80</td><td>4m</td></tr>
<tr><td>-100</td><td>10m</td></tr>
<tr><td>-128</td><td>Maximum range</td></tr>
</table>
0 - 128</td></tr>
</table>