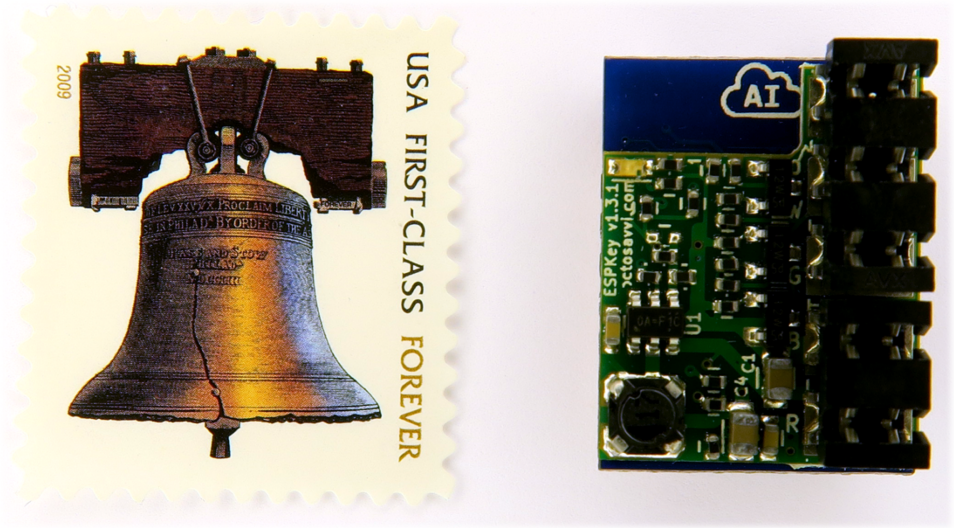
User's Guide



ESPKey
Wiegand Interception Tool

By Octosavvi

Current Revision
Document v1.0.0
Hardware v1.3.1
Firmware v128

The latest version of this document may be obtained from Red Team Tools:
http://www.redteamtools.com/espkey

Table of Contents

Introduction

What is ESPKey?

The ESPKey is an advanced implantable logic analyzer and debugging tool designed for use with any device using the Wiegand communication protocol. It has a built-in Wireless LAN communication module and can store up to 80,000 unique credential bitstreams in non-volatile memory, depending on credential bit format. The credential bitstream may be retrieved or "replayed" on demand by connecting to the built-in web interface from any mobile device or computer with a web browser.

The ESPKey is 100% transparent to both the card reader and the downstream panel.

Features

- No Battery Required
- Universal Vendor Support
- Supports Standard 5V Wiegand Signaling Protocol
- Record Wiegand Protocol Bitstream
- Transmit Wiegand Protocol Bitstream
- Log and Record Reader LED Control from Panel
- Timestamping Function (For Current Power Cycle)
- "Denial of Service" (DoS) Mode
- Supports Use of "Control" Cards to Enable DoS or Credential Replay
- 802.11 b/g/n Wi-Fi Capability with AP and Client Modes
- Supports Hidden ESSID Configuration
- Fully Customizable Web Interface
- Automatic Decoding of Common Bit Formats

Requirements

- Input Voltage Must Be Between 4.5V and 18V DC – DO NOT EXCEED 18VDC
- Interception Target Must Use Standard 5V Wiegand Signaling Protocol

Applications

Penetration Testers

- Intercept, Record, and Replay Credential Data in Transit
- Intercept, Record, and Replay Biometric Data in Transit
- Clone Recorded Bitstreams to New RFID Credentials
- Prevent Authorized Access on Demand (Denial of Service)
- Convert Standalone RFID Readers to Data Loggers

Installers

- View Binary Wiegand Bitstream from Reader
- Troubleshoot Line Quality Issues
- Test Reader Operation
- Test Panel Operation

Historical Background

What is Wiegand?

In the world of Physical Access Control Systems (PACS), “Wiegand” is a term that is used with extreme regularity, but with great inconsistency. In fact, Wiegand can refer to a physical credential type, a signaling protocol, a bit format, or even its namesake inventor. To fully understand modern PACS, it is necessary to fully understand these differences.

Wiegand, the Man

John R. Wiegand, the inventor of his namesake signaling technology, didn’t start his career as an engineer. Born in Germany in 1912, he immigrated to the United States sometime in the 1930’s to study piano and choral conducting at The Juilliard School of Music in New York City. Early on, Wiegand became interested in audio amplifiers, and in 1944 started working for a government contractor designing tape recorders. By the mid 1960’s his focus turned to research in magnetic field effects, resulting in his landmark patent in 1974 for his bistable ferromagnetic wire, now commonly referred to as “Wiegand Wire”.

Wiegand, the Wire, and Wiegand the Effect

To understand the Wiegand effect, it is important to note one of its key components, the Barkhausen effect. In 1919 Heinrich Barkhausen discovered that when a smoothly and steadily increasing magnetic field was applied to a piece of ferromagnetic material, the material became magnetized in “steps” rather than a continuous change. This unusual phenomenon has been harnessed in various practical applications such as the detection of defects in materials.

John Wiegand harnessed this curious quality in a different way. First he took Vicalloy wire 1 mil (0.010 inches) in diameter and cold-worked by twisting it and untwisting it under tension. The cold working process resulted in a wire that had a relatively hard shell, and a relatively soft center, as the center of the wire was subject to less axial work-hardening. The structure was then given permanence through age-hardening, and thus, the Wiegand wire was born.

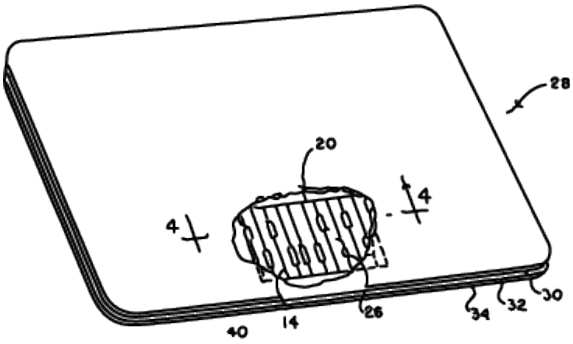
The wire’s softer core exhibited lower coercivity, while the hardened surface shell exhibited higher coercivity. When subjected to a magnetic field, the inner core is saturated first, followed by the outer shell. Once fully saturated a magnetic field of the opposite polarity is applied, causing the inner core polarity to instantly flip. This reversal causes a voltage pulse to be generated on the pickup coil. A pulse width of approximately 10 μS and 5-6V is common for most sensors.

Wigand’s discovery is doubly fascinating in part because he discovered the peculiar effects of these wires even before he had access to an oscilloscope. His lab setup used a series of magnets, coils, and speakers that allowed him to listen to the wires, which he often referred with a female pronoun. Somewhat endearingly, Wiegand would say that his wires “sung” to him.

Over the next four decades, his peculiar discovery would find its way into nearly every facet of modern life. Since the wires themselves never degraded, and the pickup sensors could be used at a distance from the wires themselves, they never wore down or were subject to mechanical stress. This made them ideal for numerous industrial applications, and were used in a wide variety of linear and rotary encoders, including utility meters, anti-lock braking systems, speed sensors, positional indicators, casino chips, and countless other applications including…Physical Access Control Systems.

Wiegand, the Credential

As the utility of Wiegand wires became more clear, it wasn't long before they were used for access control purposes. While magnetic stripe was subject to mechanical wear, abrasion, and corruption by external magnetic fields, Wiegand wires were not subject to any of the same stressors. Taking advantage of this, Wiegand sandwiched a series of wires between plastic and the Wiegand swipe credential was born.



Wiegand wires were physically placed in differing positions on the credential depending on the corresponding 0 or 1 bit that they represented. When the credential was swiped through the card reader, two different pickup coils would detect 0 or 1 bits depending on the position of the wires on the card. Thus, the 26 wires present on a traditional Wiegand swipe credential represented 26 bits of credential data to be used by the door controller for access control.

Wiegand, the Signaling Protocol

Wiegand readers use a simple two-wire signaling protocol consisting of two 5V data lines, one for carrying 0 bits, and another for carrying 1 bits, commonly referred to as D0 and D1, respectively. D0 and D1 are both held high at 5V by default when there are no bits in transit, and when credential data is transmitted the corresponding D0 and D1 lines are pulsed low to 0V with a pulse width between 20 μS and 200 μS. The time between pulses can vary significantly between different readers, and the specification allows anywhere between 200 μS and 20,000 μS between pulses.

In 1996 the Wiegand specification was formally adopted by the Security Industry Association (SIA) as thus, what started first as a proprietary protocol for a new and revolutionary product became the de facto standard for modern access control.

Wiegand, the Data Format

Complicating Wiegand matters further is the Wiegand *data format* or *bit format*. The data format refers to how credential data is presented on the wire. Specifically, it may refer to how many bits are used, and how those bits are encoded, and how parity is calculated.

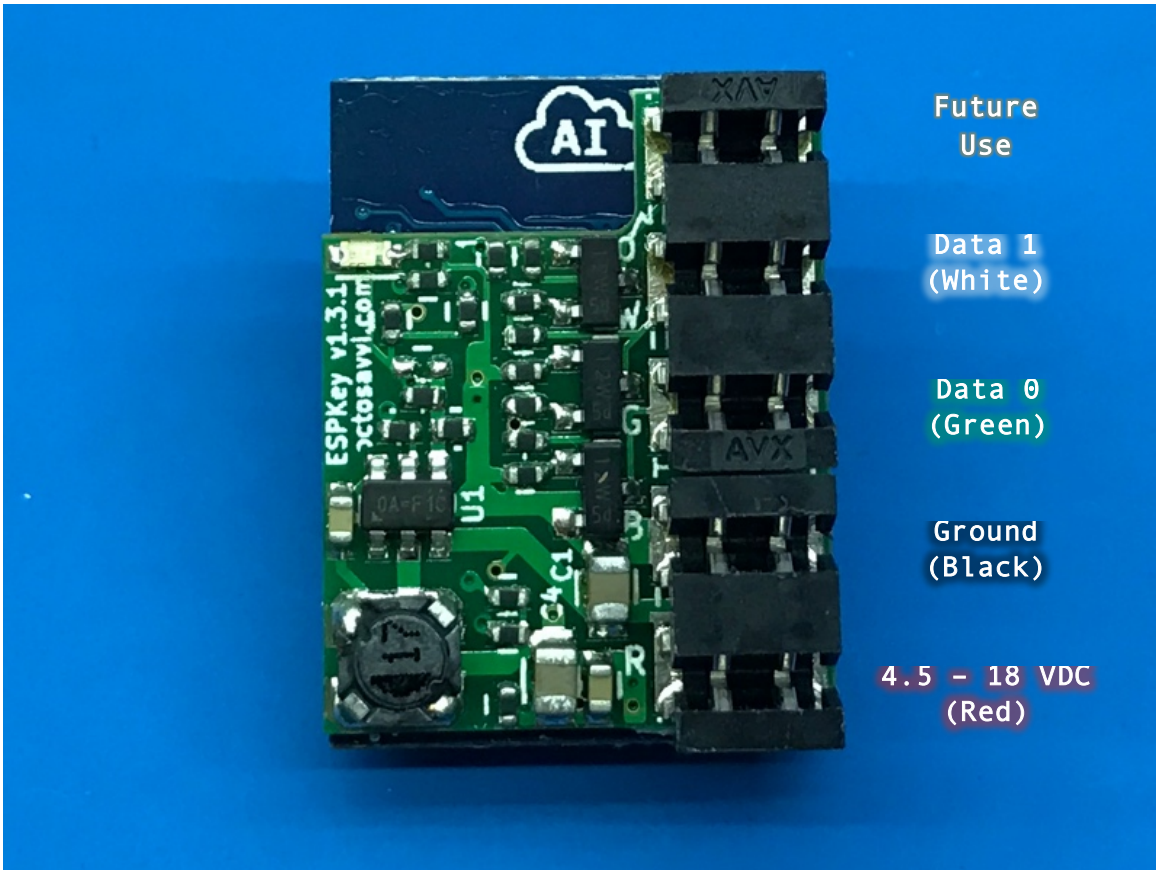
The most common format in use is the standard 26-bit format defined in the 1996 SIA specification, referred to by HID under ordering code H10301, but not all data formats are standardized. Several dozen proprietary and non-standard bit formats exist today, and a “37 bit format” from one vendor may use a different encoding from another vendor advertising the same bit length.

It is important to note that the ESPKey’s recording and replay abilities are format-agnostic, meaning the data format used by the reader or the credential does not impact the operation of the ESPKey. The binary bitstream is captured in its original form, and retransmitted bit for bit upon demand.

The data format may become more important for parsing out the Facility Code and Card Number if a user wishes to use the credential data to encode the data back onto a physical card.

Using ESPKey

Pre-Deployment



Pre-Flight Check

Before deployment, it is necessary for the operator to familiarize oneself with the ESPKey's hardware layout. Take note of the device pinout above. The PCB has small R, B, G, and W markings that indicate Red (VDC), Black (Ground), Green (Data 0), and White (Data 1), respectively. **It is important to understand that the wire colors are commonly accepted guidelines, but some installations may use alternate colors.**

The ESPKey uses a series of five Insulation Displacement Connectors, or IDC's. With the use of a matched crimping tool, these connectors allow the connection of wires to the device without needing to strip off insulation first. The IDC will cut the insulation on the wire and create an air-tight connection with the copper wire in one step. The connectors installed on the ESPKey are designed for use with wire sizes between 18 and 24 AWG.

Powering the ESPKey

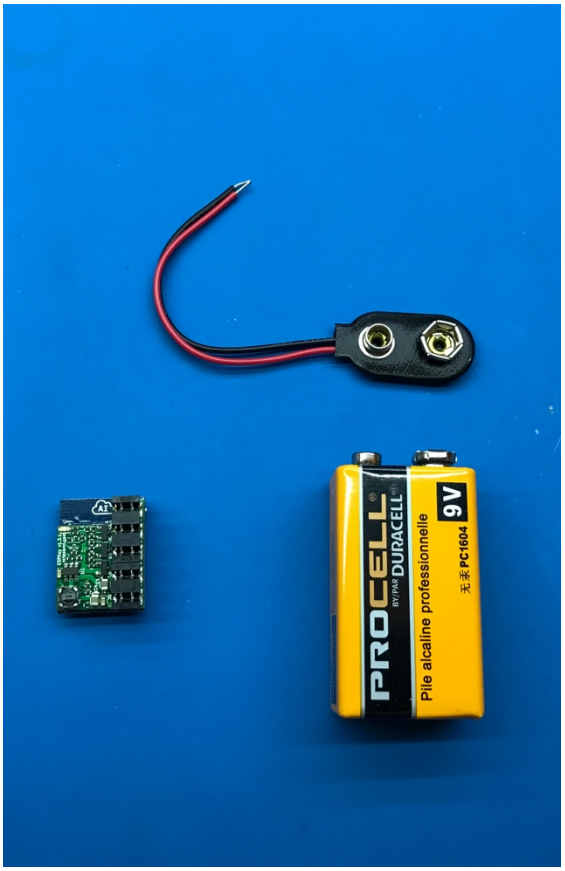
To verify that the ESPKey is operational and to ensure it is properly configured, it is necessary to power up the ESPKey prior to field deployment.

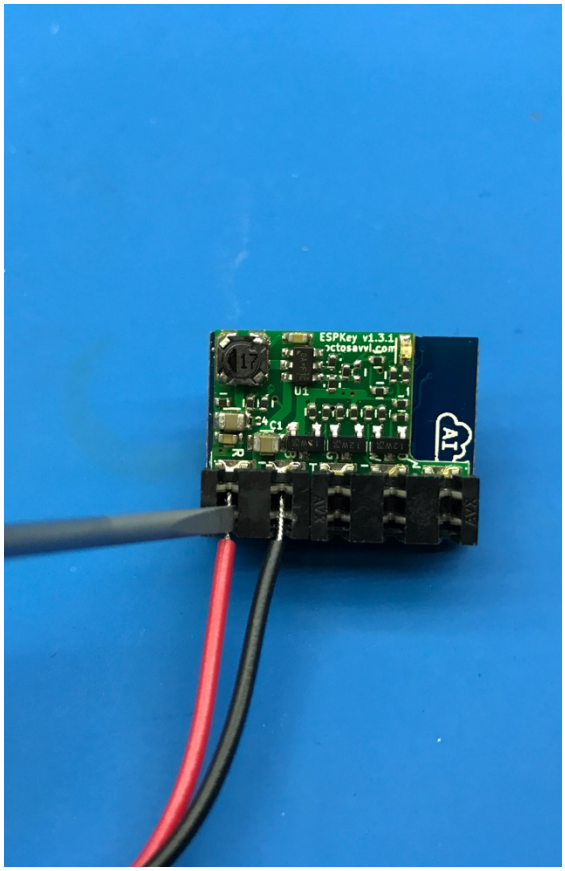
ESPKey requires a minimum of 4.5VDC to operate, and supports up to 18VDC input.

IT IS RECOMMENDED THAT REGULATED POWER SUPPLIES BE USED IN CONJUNCTION WITH ESPKEY.

ESPKEY DOES NOT HAVE REVERSE-POLARITY PROTECTION. INCORRECT CONNECTION OF POWER TO ESPKEY WILL DAMAGE ESPKEY.

The simplest and most accessible power source in many cases will be a 9-volt battery, as shown below:



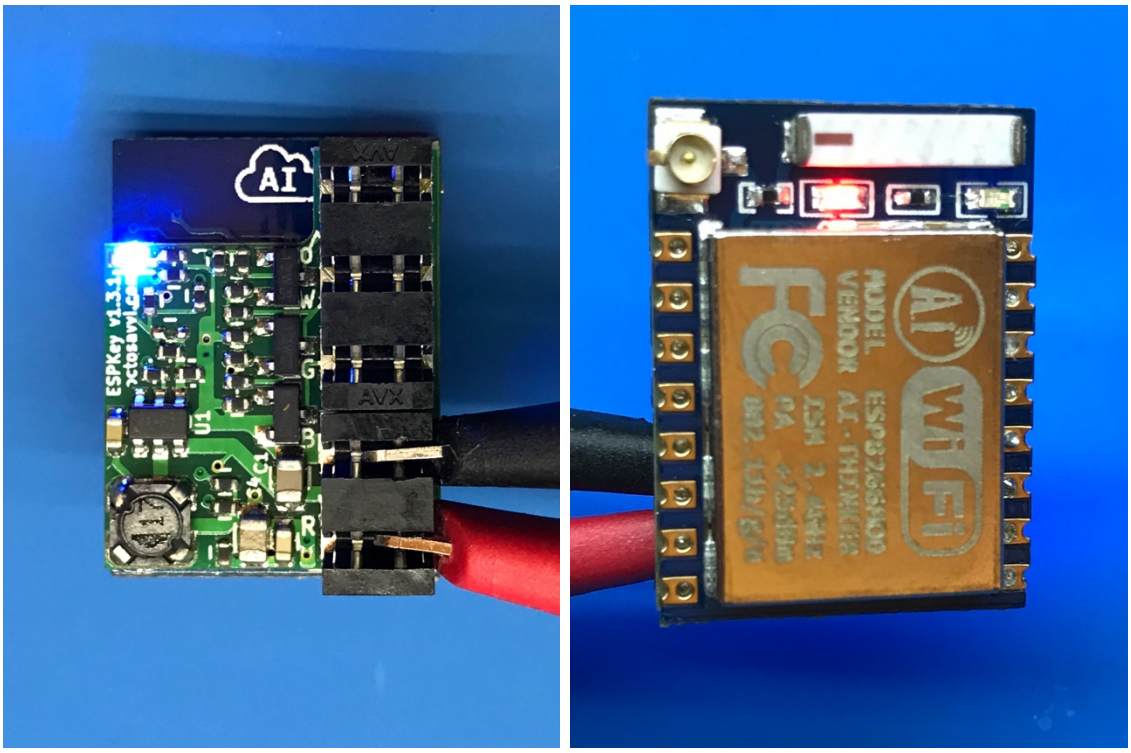


If the wire ends are stripped, it may be possible to gently press the wires into the IDC connectors using a small flat blade screwdriver.

If a 9-volt battery is not available, a standard USB cable may also be cut in half, and may be used to provide the ESPKey with 5VDC from a standard USB power source. Note that USB cables also contain green and white wires for data transmission. **Do not connect these wires to the ESPKey!**

Connecting to the ESPKey User Interface

Once the ESPKey is connected to power, two LED's will illuminate on the tool. A blue LED on the side with the IDC, and a red LED on the side with the metal RF cage.

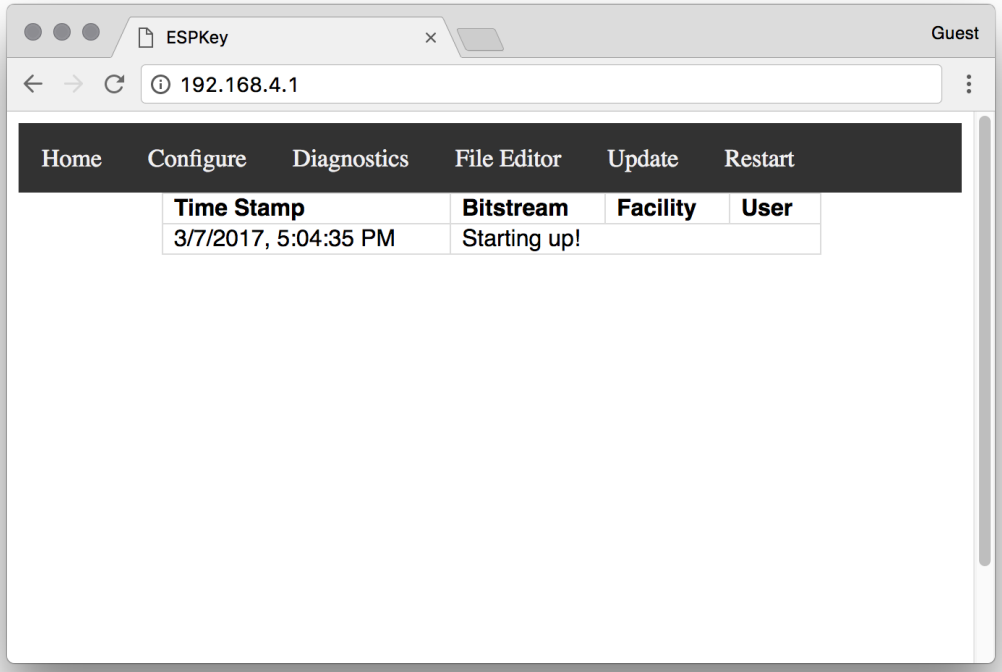


The ESPKey contains a built-in access point and uses Wi-Fi to provide a web-based user interface for interaction with the tool. In its default configuration, the ESPKey will broadcast in AP mode using the following configuration:

SSID:	ESPKey-config
Security:	WPA / WPA2 Personal
WPA Key:	accessgranted
Gateway:	192.168.4.1
UI URL:	http://192.168.4.1/ http://espkey.local/

To connect to the User Interface, use a smartphone, tablet, notebook, or desktop computer with Wi-Fi to connect to the “ESPKey-config” access point. Once connected, use a web browser to navigate to http://192.168.4.1/ Most modern devices will allow the tool to be accessed via the mDNS service as well, http://espkey.local/

If the following page loads, it is in indicator that the ESPKey is operational.

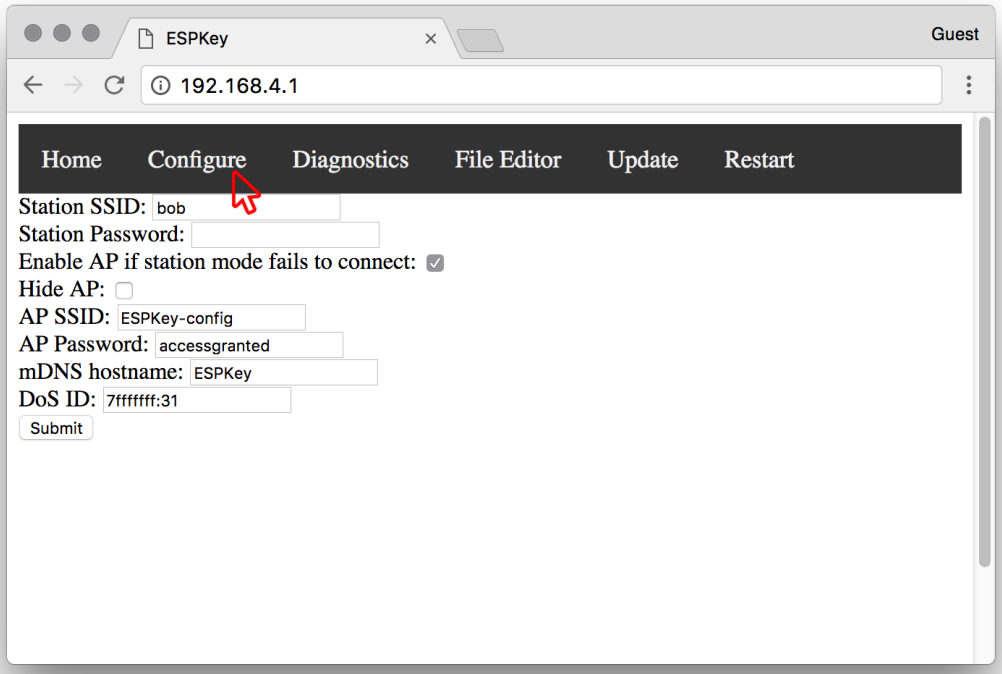


The ESPKey may now be configured pursuant to the needs of the operator.

DEPLOYMENT OF ESPKEY AGAINST A TARGET IN ITS DEFAULT CONFIGURATION MAY RESULT IN ACCESS BY AN UNAUTHORIZED OR UNAPPROVED THIRD PARTY

Configuring the ESPKey

Correct configuration of the ESPKey is necessary to ensure reliable operation in the field. Incorrect configuration of the tool may result in an inability of the field operator to remotely control the ESPKey.



By default, the ESPKey will first attempt to operate in client mode. In client mode, the ESPKey will connect to the Wi-Fi station identified in the “Station SSID” field. If the specified station is not available, the ESPKey will revert to AP mode, and will broadcast an SSID as specified in “AP SSID” using the key specified in “AP Password”. If differing operation is desired, the ESPKey should be configured accordingly prior to field deployment.

DEPLOYMENT OF ESPKEY AGAINST A TARGET IN ITS DEFAULT CONFIGURATION MAY RESULT IN ACCESS BY AN UNAUTHORIZED OR UNAPPROVED THIRD PARTY

ESPKey settings may be configured in consideration of the following chart:

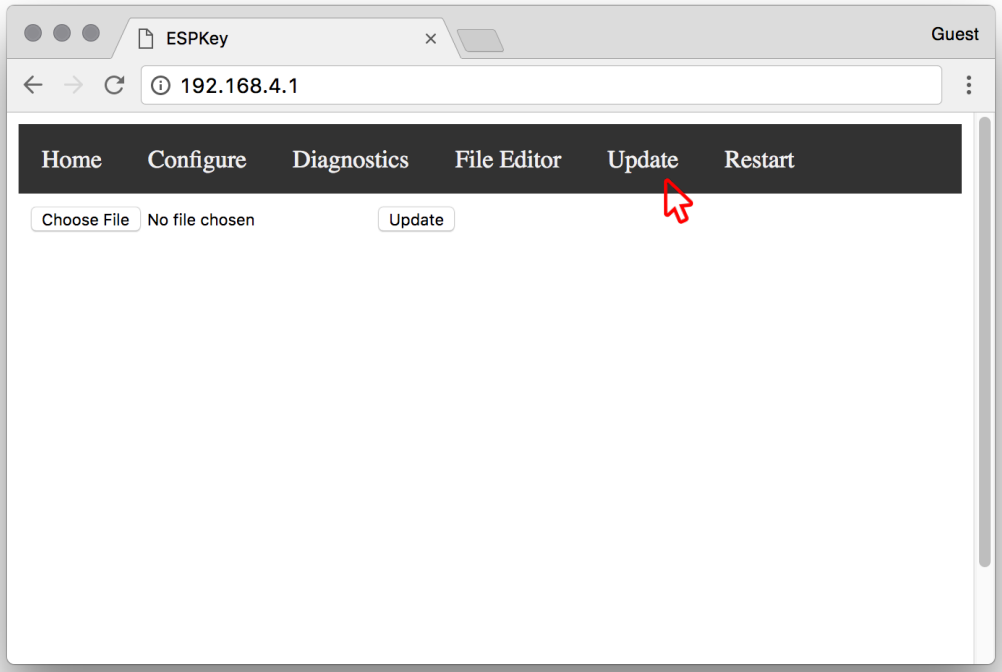
Parameter	Description	Default
Station SSID	This is the SSID the ESPKey will attempt to connect to upon power-up.	bob
Station Password	If the SSID the ESPKey will connect to requires a password, it should be specified here.	
Enable AP if Station Mode Fails to Connect	Upon power-on the ESPKey will connect to the SSID specified above. If the specified station is unavailable and this box is checked, the ESPKey will broadcast in AP mode using the settings specified. Once the ESPKey begins broadcasting in AP mode it will not attempt client mode again until the ESPKey is restarted.	Enabled
Hide AP	If this option is selected the ESPKey will use a hidden SSID when broadcasting in AP mode.	Disabled
AP SSID	This parameter specifies the SSID while broadcasting in AP mode.	ESPKey-config
AP Password	This field specifies the WPA2 password while broadcasting in AP mode.	accessgranted
mDNS Hostname	This field will specify the hostname that will be broadcast via the mDNS service. On most modern devices <mdnshostname>.local may be used in place of the IP address to connect to the ESPKey UI.	ESPKey
DoS ID	If the ESPKey detects a credential read with a value matching this field, the tool will enter Denial-of-Service (DoS) mode. Please refer to DoS section in this manual for more information. The format of this field is <card value in hex>:<number of bits>	7fffffff:31

Updating Firmware

The ESPKey supports Over-the-Air (OTA) firmware updates for the addition of new features and bug fixes. The current firmware version of the ESPKey may be viewed by navigating to http://<IP or mDNS>/version

To check to see if an ESPKey firmware update is available, please visit http://www.redteamtools.com/espkey or contact espkey@redteamtools.com

The firmware may be updated via the web UI, and may be accessed by clicking the "Update" tab or by navigating to http://<IP or mDNS>/update



UPDATING ESPKEY FIRMWARE WILL RESULT IN LOSS OF LOGGED CREDENTIALS

While powering the ESPKey using a reliable power source select the binary using the file browser. Once a selection is made, clicking the "Update" button will begin the firmware upgrade process, which will typically complete within 5-10 seconds. Once the update is complete, an "Update Success!" notification will be displayed and the ESPKey will reboot automatically.

In some cases when major changes are made to the firmware, the UI files will need to be re-uploaded. In such cases, connecting to the ESPKey will result in a "File Not Found" error. See "Updating UI" in the next section for further information.

Updating UI

Due to the design of the ESPKey memory, in some cases major firmware updates will corrupt or damage the area of the file system storing the UI. In these cases, the UI files will have to be re-uploaded from a desktop or notebook computer using the ui-update.sh script.

The latest version of the UI update script may be found at http://www.redteamtools.com/espkey

The current version of the script must be run from macOS or Linux system at this time. Windows instructions will be released at a future time.

To update the UI following a firmware update, perform the following actions:

1. Retrieve the latest version of firmware and ui-update.sh script.
2. Update firmware to latest version using instructions in “Firmware Update” section.
3. Connect to the ESPKey via Wi-Fi, and run ui-update.sh script from client system.
4. If the ESPKey has an IP address that is different from the default configuration, it maybe be supplied as a command argument following the script:

```
/ui-update.sh 192.168.4.1
```

Deploying ESPKey

Deployment Precautions

ESPKey should be considered a sensitive electronic device. Care should be taken not to short ESPKey against metal surfaces during operation, such as housings, junction boxes, and grounding wires. If necessary, electrical tape should be used to insulate the ESPKey from such surfaces.

ESPKey will be damaged by incorrect or improper installation. It is critical for the operator to familiarize oneself with the layout of the IDC connectors prior to deployment.

ESPKey requires a minimum of 4.5VDC to operate, and supports up to 18VDC input. While 99% of readers in the field will conform to these specifications, some medium and long range readers may be connected to 24V power sources. **Connecting ESPKey directly to such sources will permanently damage the ESPKey. If target line voltage is unknown, it should be checked with a voltmeter.** If it is necessary to connect ESPKey to a reader exceeding these specifications, an external battery or alternate power source must be used.

ESPKEY DOES NOT HAVE REVERSE-POLARITY PROTECTION. INCORRECT CONNECTION OF POWER TO ESPKEY WILL DAMAGE ESPKEY.

All ESPKey transmissions will be received by the downstream control panel. In most cases, this may result in log entries on the target system that may indicate compromise. Field operators should plan their use of the tool accordingly.

Reader Retention Mechanisms

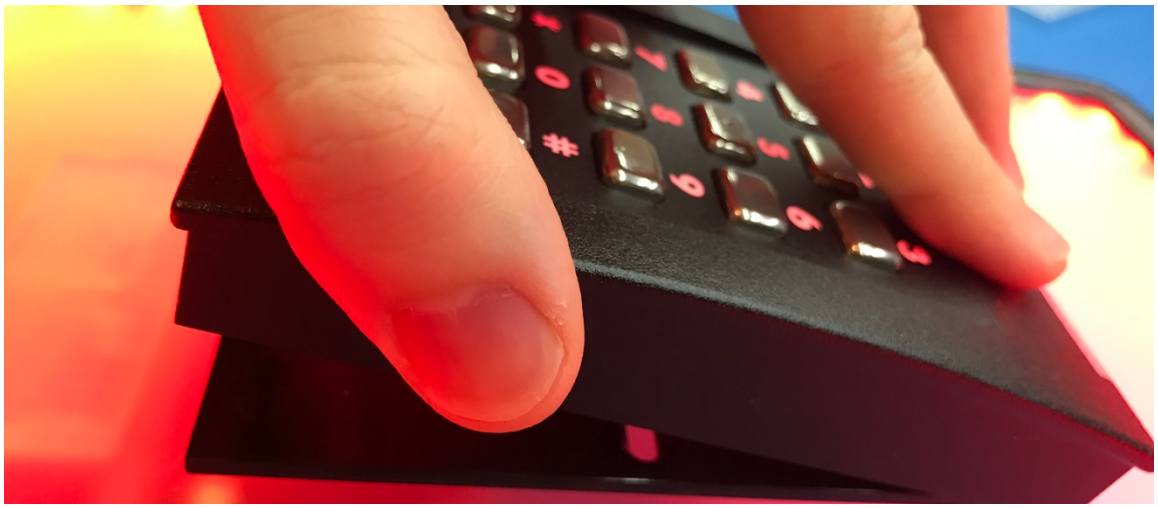
While the ESPKey is reader-agnostic, the method for gaining access to the reader wiring may differ between installations. Although it is impossible to cover every possible configuration, several common configurations are outlined in this manual.

In general, most RFID readers encountered in the field will use one of several common retention mechanisms to anchor the reader to the wall.

Single Point Retention

A reader using this retention method will usually use a single fastener, often on the bottom edge of the reader, to secure the edge of the reader against the wall.

Once the fastener is removed, the reader may usually be removed by tilting it away from the wall using the side opposite the fastener as a hinge.



Hidden Panel Thru-Hole Retention

Some readers are mounted using thru-hole fasteners to mount the reader directly to the wall, but use a hidden panel or sticker to conceal the fasteners for aesthetic reasons.

Some hidden panels may have tool insertion points to aid in removal, while others may require prying at an edge. In the example above, a small flathead screwdriver is used to lift out the screw cover plate. If the fasteners for the reader are not clearly visible, it is strongly advised to consult the installation documentation for the target reader prior to deployment of the ESPKey tool.

Multi-Point Edge Retention

Larger RFID readers, such as those used in parking lots, may at times use multiple fasteners on the edge of the reader for retention. In such cases, all of the fasteners must be removed in order to gain access to the required wiring.

Tamper Resistant Installations

Tamper-Resistant Fasteners

While rare, readers may also be installed and secured using tamper-resistant fasteners.

In such cases a “security bit set” is useful for field deployment, but care should be taken to not to damage a tamper-resistant fastener with the use of an incorrect bit.

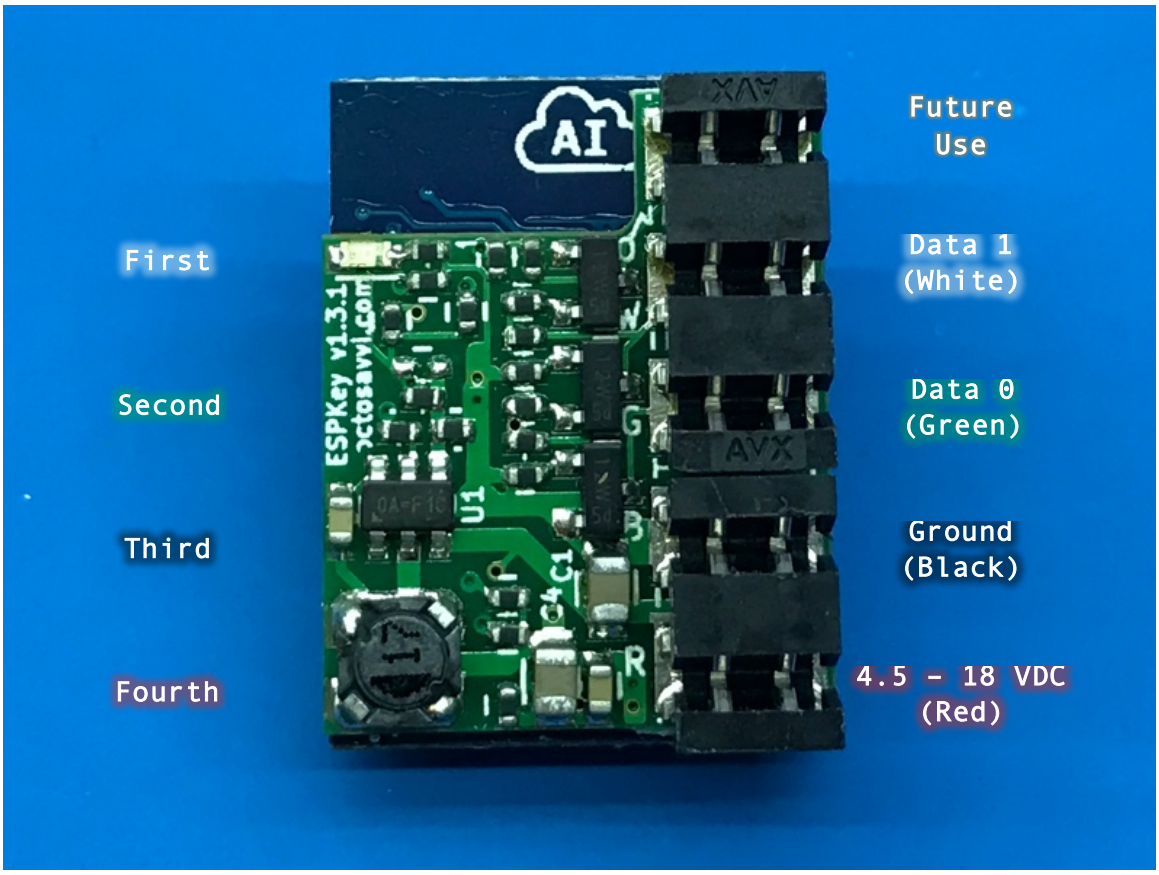
Tamper Detection Mechanisms

While rare, most modern readers have hardware support for a “tamper detection” feature of some kind. If such sensor is wired to the control panel for monitoring, it must be mitigated during ESPKey deployment to prevent detection of the field operator.

Each model of reader may employ a different form of tamper detection, and it is recommended for the operator to review installation documentation for target readers for the location and operation of relevant sensors.

Connecting ESPKey to Target

The ESPKey tool may be sensitive to sudden power surges, and it is important to follow the proper connection order to reduce the risk of damage during field deployment.



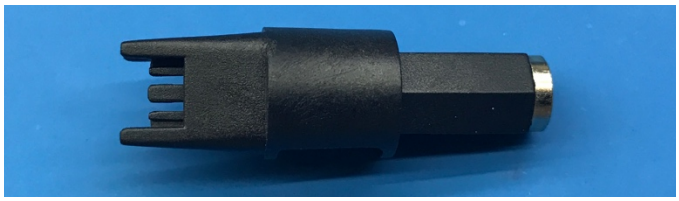
The ESPKey uses a series of five Insulation Displacement Connectors, or IDC's. With the use of a matched crimping tool, these connectors allow the connection of wires to the device without needing to strip off insulation first. The IDC will cut the insulation on the wire and create an air-tight connection with the copper wire in one step. The connectors installed on the ESPKey are designed for use with wire sizes between 18 and 24 AWG.

Take note of the device pinout above. The PCB has small R, B, G, and W markings that indicate Red (VDC), Black (Ground), Green (Data 0), and White (Data 1), respectively. **It is important to understand that the wire colors are commonly accepted guidelines, but some installations may use alternate colors.**

Once the reader wiring is located, the wires should be connected in the following order:

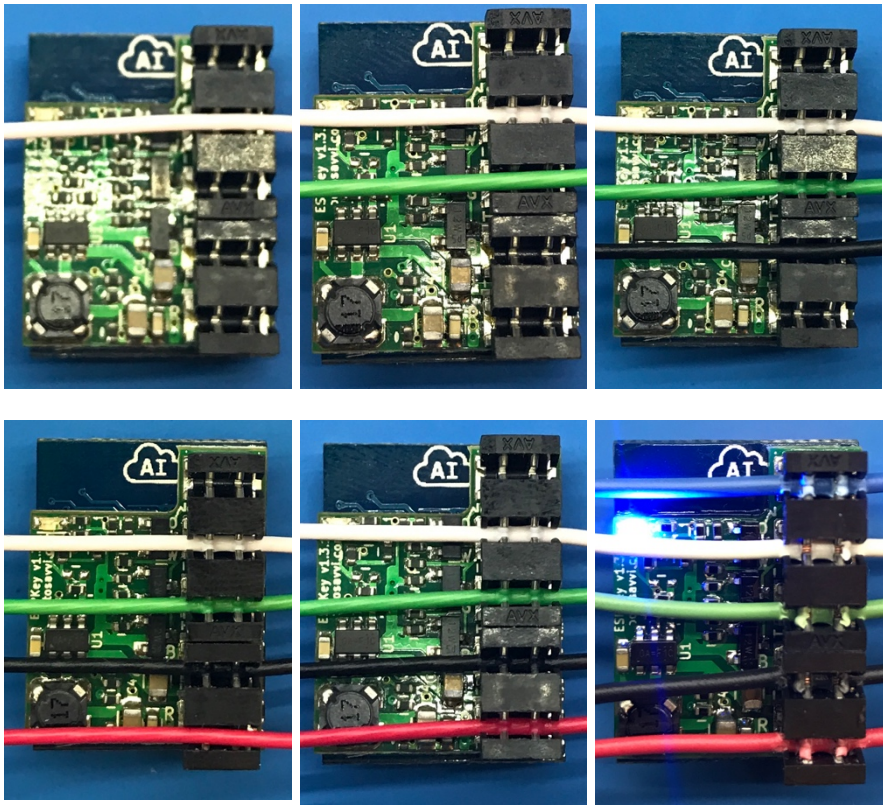
1. Data 1 (White)
2. Data 0 (Green)
3. Ground (Black)
4. 4.5-18VDC (Red)

AVX Punch-down Tool Usage



To prevent damage to the ESPKey IDC, the correct AVX punch-down tool is required. Replacement or extra punch-down tools may be ordered from http://www.redteamtools.com/espkey or from major electronics parts distributors using AVX P/N 069176701602000.

The punch-down tool is a standard ¼ inch hex bit with a magnetic bottom plate, and may be used in conjunction with common bit handles designed for ¼ inch bits.

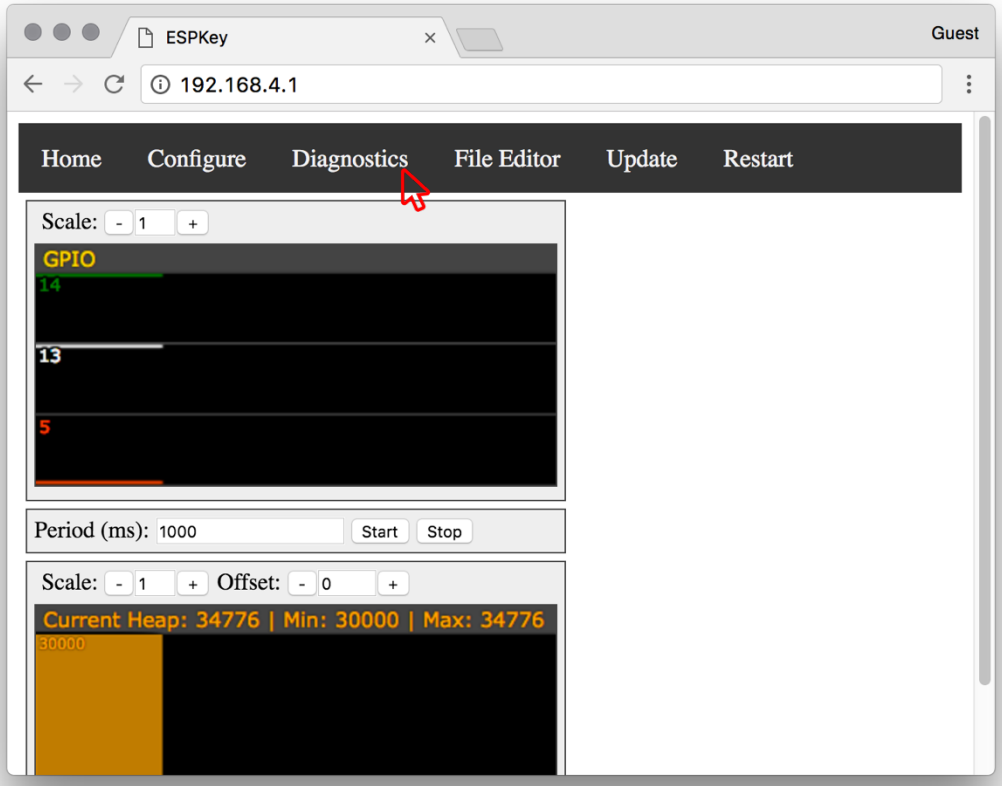


To ensure optimal contact with the IDC, perform the following steps during punch-down:

1. Place the ESPKey should be placed against a hard, flat surface such as the reader mounting plate or wall.
2. Hold the punch-down tool perpendicular to the surface and align the tool with the wire and connector to be punched.
3. Firmly press the wire into the IDC using the punch-down tool until the wire is fully seated. In most cases a gentle “click” or impact may be felt in the tool as the wire is seated.
4. Visually check the wire to ensure that insulation appears to be pierced cut and that wire is fully seated.
5. Observe that blue power LED is illuminated to the left of the IDC.

Diagnostic View

The ESPKey UI contains a rudimentary diagnostic display that may be used to confirm positive connection to Wiegand data lines.



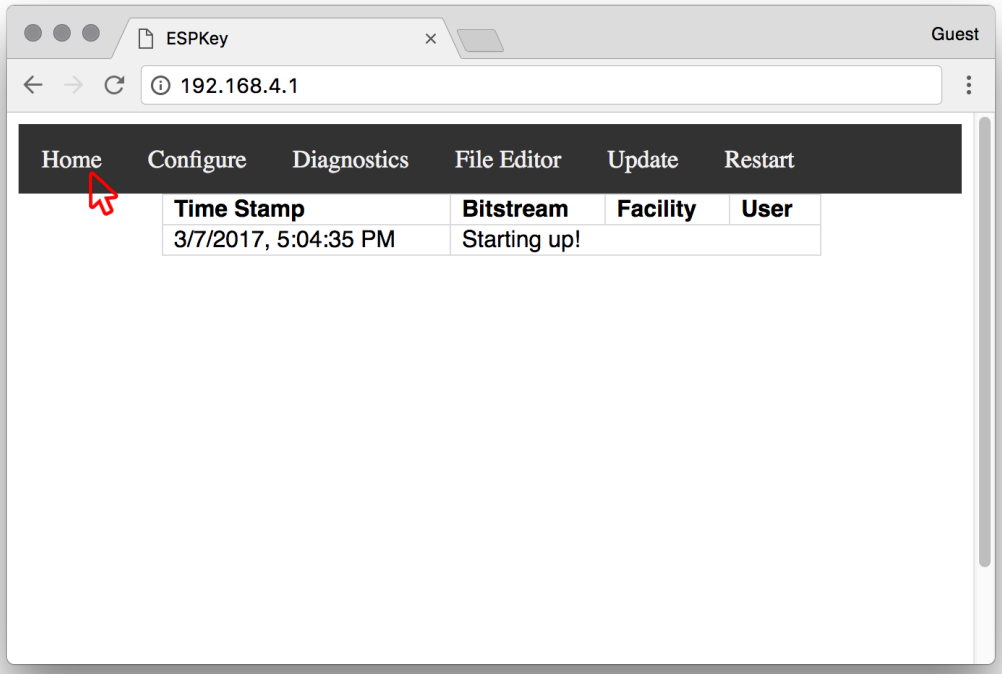
When properly connected, both D0 and D1 lines should be represented by solid “high” lines on the graph. These lines are represented by GPIO pins 14 and 13, respectively.

Connecting to ESPKey via Wi-Fi

Once the ESPKey is connected to power, the tool will power up and immediately begin logging operations. It is recommended to test Wi-Fi functionality of the ESPKey in the field once the tool is installed to ensure installation was successful.

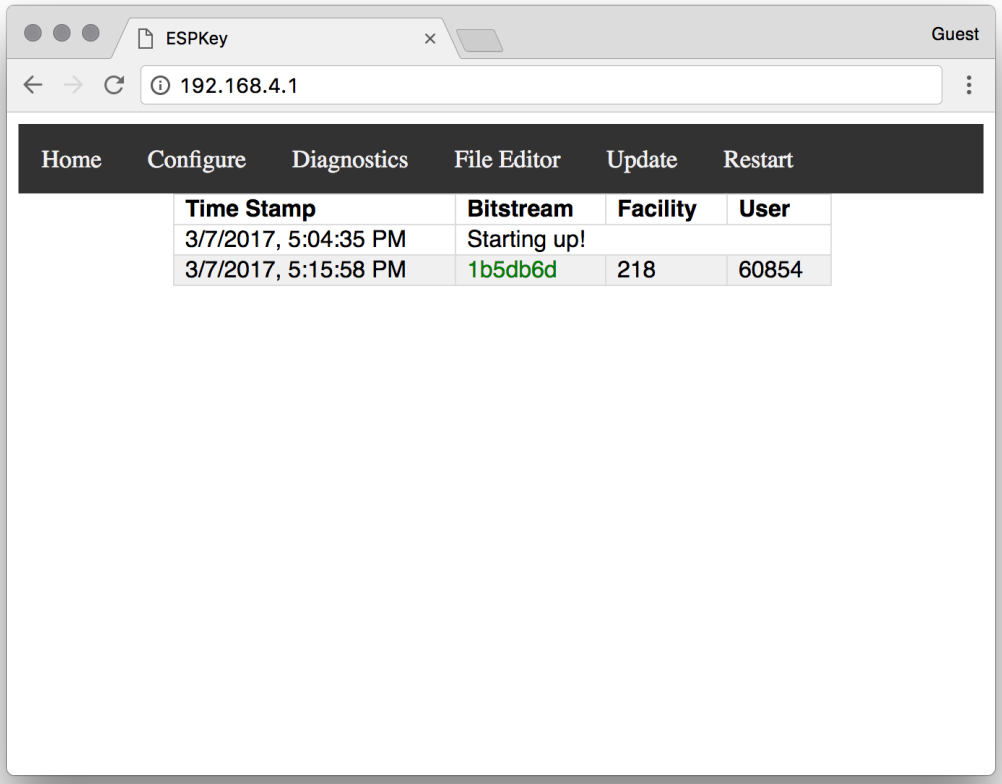
The specifics of connecting to the deployed ESPKey will depend on the configuration settings of the tool defined during pre-deployment configuration. In the default configuration ESPKey will attempt to connect to the station SSID specified in the configuration. If the station is unavailable, ESPKey will broadcast the SSID defined in AP configuration.

Without any credentials logged, the welcome screen for ESPKey Tool should look similar to the screen below.



Viewing Credential Log

The credential log will be displayed by default when connecting to the ESPKey web UI. It is important to understand that ESPKey does not differentiate between transmissions originating from the reader and transmissions originating from ESPKey. There is no way to discriminate between such transmissions at this time.



The activity log may be interpreted in consideration of the following:

Time Stamp

The ESPKey does not contain a Real-Time Clock (RTC). Instead, the ESPKey begins counting time from the moment it is powered on, and the time stamp is calculated relative to the local system clock of the web browser being used. Accordingly, the ESPKey is unable to calculate the time stamp for activity logged prior to the current power cycle. If such activity exists, the stamp will indicate a question mark (?) in lieu of a time.

Bitstream

The binary Wiegand bitstream is recorded and displayed in hexadecimal format, including parity bits, if applicable. This is the binary data as it is represented on the wire, and is identical to the data interpreted by the control panel.

Clicking on the “Bitstream” value will open the retransmission toolbox for the selected credential. See “Credential Replay / Retransmission” for further information.

Facility and User Code

As has been mentioned earlier in the beginning of this document, Wiegand data formats can be widely varied, and interpretation of the binary data and its translation to “Facility” and “User” codes is often proprietary.

When possible, ESPKey will attempt decode the bitstream according to the detected data format. **Data format detection is imprecise and should not be considered reliable.** It is not necessary to decode the data format to execute replay attacks against the target system.

At time of writing the ESPKey UI will automatically attempt to decode the following data formats:

- Standard 26-Bit Data Format (H10301)

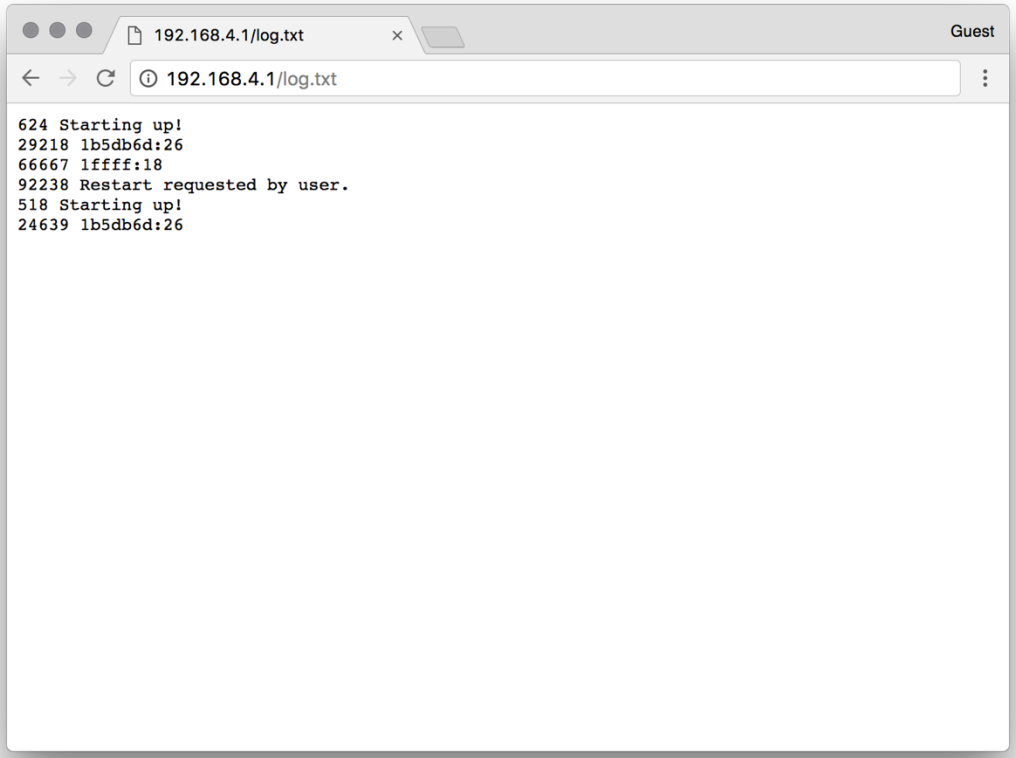
As of firmware version 128 ESPKey will highlight successfully decoded bitstreams in green, while unknown bitstreams will be shown in red. This functionality will likely change in future firmware versions.

Additional formats may be added in the future, but some proprietary formats will not be added due to lack of information available regarding the format.

If you have information regarding a format you would like to see added, please forward the request to espkey@redteamtools.com

Viewing the Plaintext Credential Log

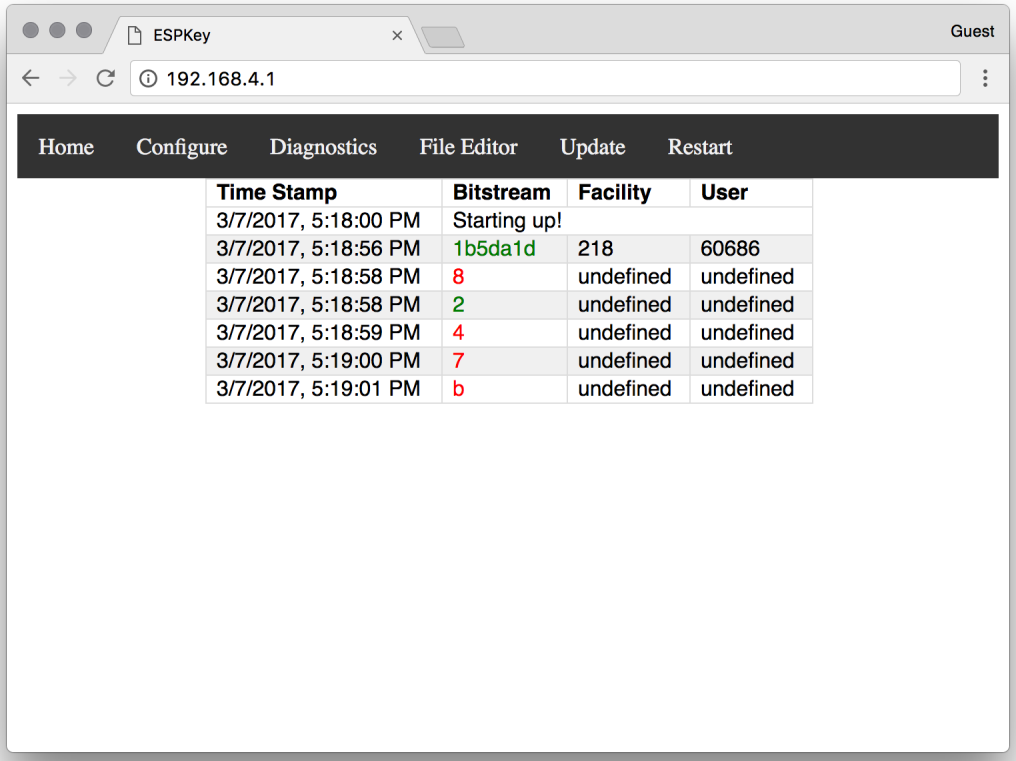
If desired, the unprocessed log.txt file may be accessed directly by connecting to http://<IP or mDNS>/log.txt


192.168.4.1/log.txt
Guest
192.168.4.1/log.txt
624 Starting up!
29218 1b5db6d:26
66667 1ffff:18
92238 Restart requested by user.
518 Starting up!
24639 1b5db6d:26

PIN Entry and Two-Factor Authentication

Some PACS are configured to require two-factor authentication for specific areas or users. In normal operation, the authorized user will present a physical credential, followed by PIN-entry, or begin with PIN-entry and follow with a credential read.

While some readers may allow for “local PIN verify” modes, where the PIN is authenticated against data stored in the RFID credential, most systems perform remote-verification by transmitting PIN data from the reader to the control panel via the Wiegand signaling protocol.



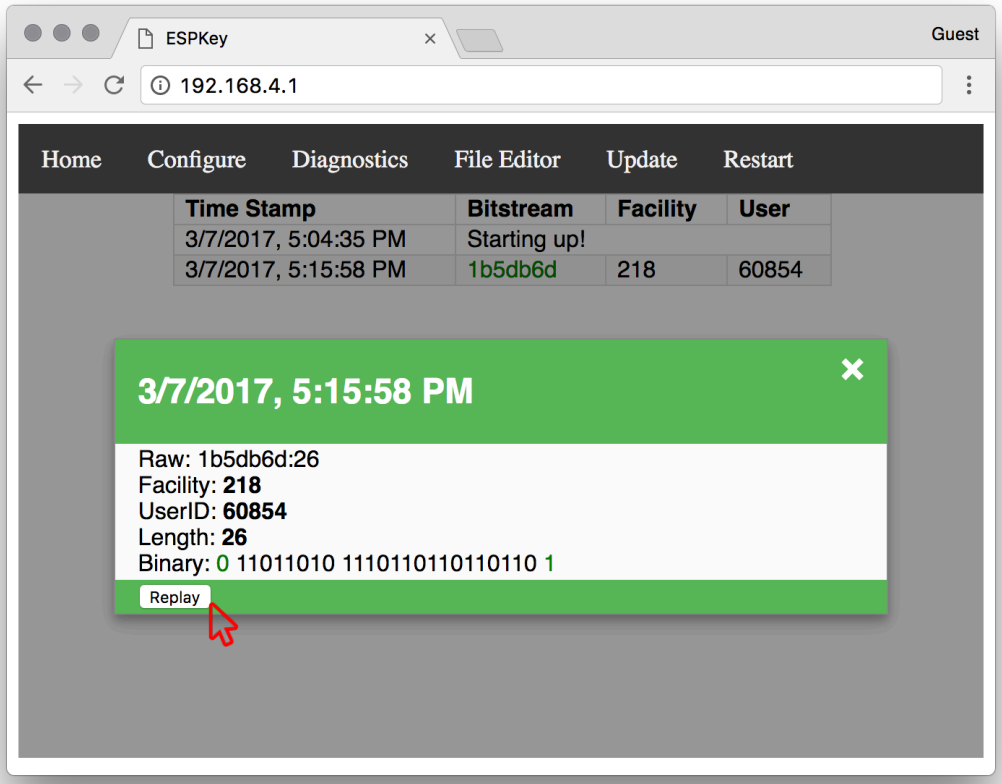
Time Stamp	Bitstream	Facility	User
3/7/2017, 5:18:00 PM	Starting up!		
3/7/2017, 5:18:56 PM	1b5da1d	218	60686
3/7/2017, 5:18:58 PM	8	undefined	undefined
3/7/2017, 5:18:58 PM	2	undefined	undefined
3/7/2017, 5:18:59 PM	4	undefined	undefined
3/7/2017, 5:19:00 PM	7	undefined	undefined
3/7/2017, 5:19:01 PM	b	undefined	undefined

As with other Wiegand data formats, PIN data formats may also be inconsistent. Simple PIN formats will be displayed plainly under the “Bitstream” column, while other proprietary formats may require an alternate interpretation. In such cases, it may be helpful for the operator to press each button once and in order, and compare the resultant activity log to the input to correlate the data.

In the above example, the time stamp reflects that shortly after the *1b5da1d* credential was presented, a PIN of “8247” was entered into the keypad, followed by ‘#’, which is represented as ‘b’ in hexadecimal in the activity log.

Credential Replay / Retransmission

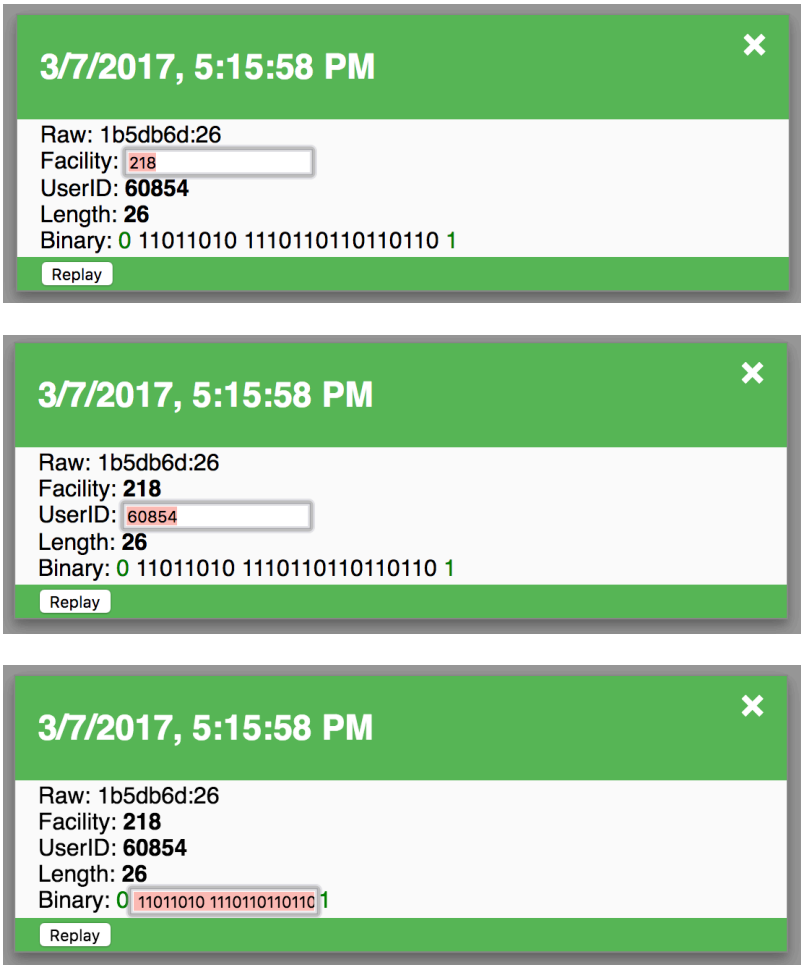
ESPKey allows for several methods of credential replay or retransmission. The simplest and most direct method is accessed by selecting the credential from the “Bitstream” column in the activity log, bringing up the retransmission toolbox.



Clicking on the “Replay” button will immediately retransmit the selected credential to the control panel. If the associated user is currently authorized for access, the door controller will release the door and the reader LED control may indicate that access has been granted.

Credential Modification Prior to Retransmission

In many installations user credentials may be issues sequentially, and in some cases it may be helpful to the operator to modify a recorded credential prior to retransmission. The ESPKey allows for this functionality via the retransmission toolbox UI.



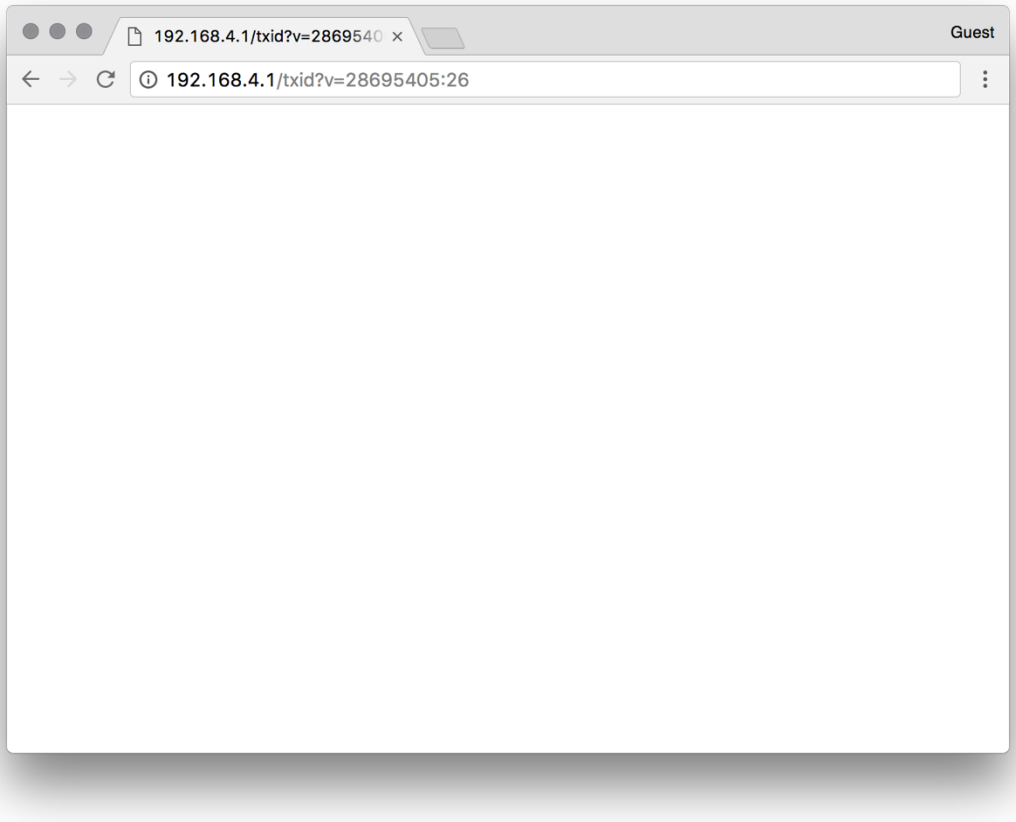
While the retransmission toolbox is open, clicking on the Facility, UserID, or Binary fields allows the operator to modify the credential data prior to retransmission. **Note that if the Facility code or UserID fields are changed, the Binary field must not be adjusted.**

Transmitting an Arbitrary ID

The ESPKey allows for the field operator to transmit a binary, arbitrary Wiegand bitstream without modifying an existing entry. The bitstream to be transmitted must first be converted to **decimal** (base-10) format and may be passed as a variable in the URL using the following format:

http://<IP or mDNS>/txid?v=<BitstreamInDecimal>:<BitLength>

In the following example, a field operator needs to transmit a binary bitstream of "01101101011101101101101101" it must be first converted to decimal format. Following the conversion the bitstream is represented as "28695405", and has a **binary** bit length of 26 bits.



ESPKey will not provide any visual confirmation of a manual transmission and will only load a blank page following transmission. To confirm retransmission, the activity log may be reviewed.

Denial-of-Service Mode

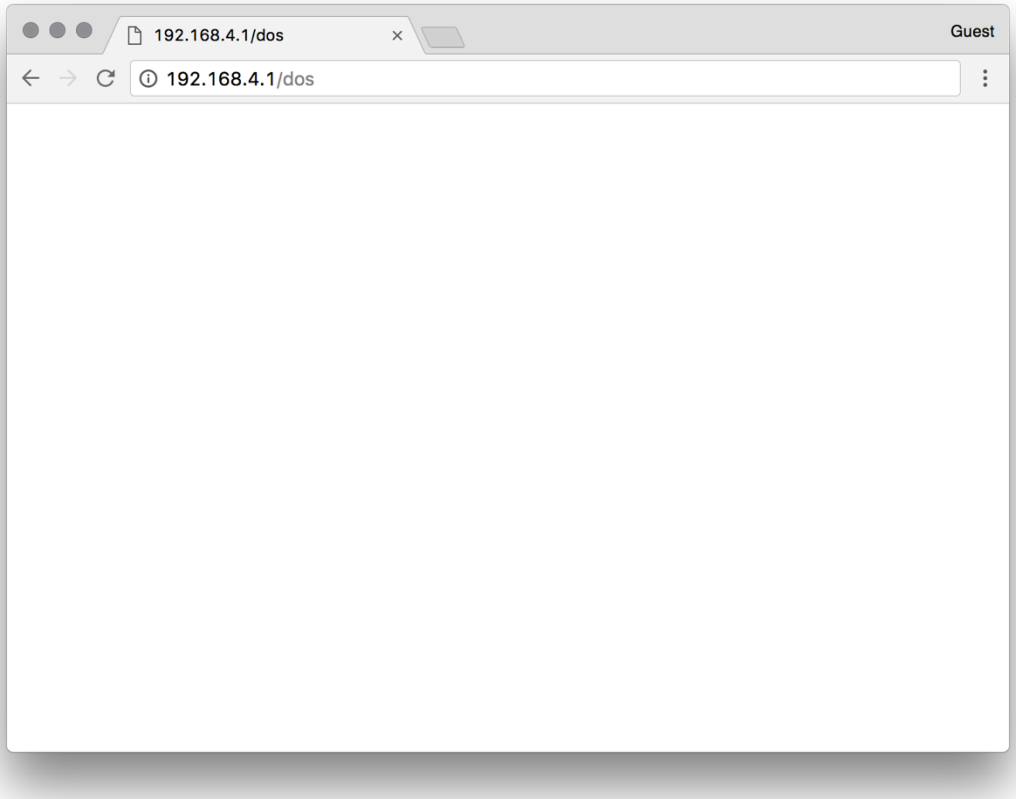
The ESPKey supports a Denial-of-Service (DoS) mode that enables a field operator to temporarily interrupt transmissions between the reader and the control panel. When enabled, ESPKey will monitor the D0 and D1 data lines for activity, and add additional spurious data bits to the credential to invalidate the transmission.

DoS mode will prevent all normal credential transmission while enabled, including transmission attempts made from the ESPKey UI.

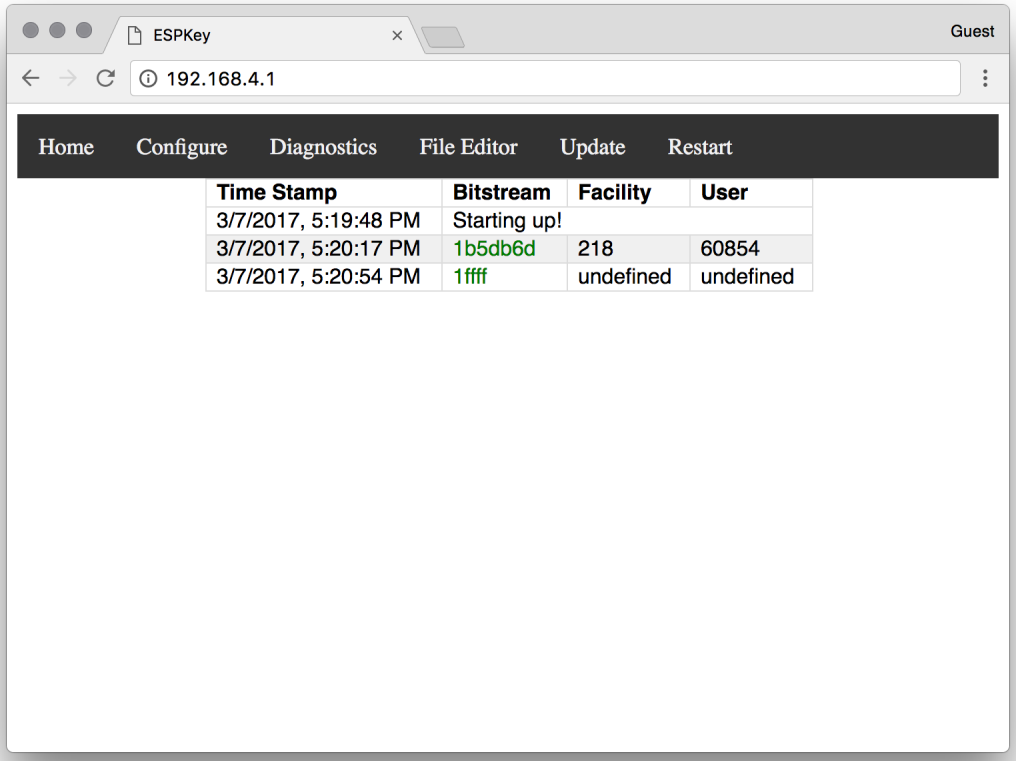
Activating DoS via URL Request

DoS mode may be activated by the loading of the following URL:

http://<IP or mDNS>/dos



Once activated, additional credential transmissions will be detected by ESPKey and modified in-transit. Modified credentials will be listed as “1ffff” in the transaction log.



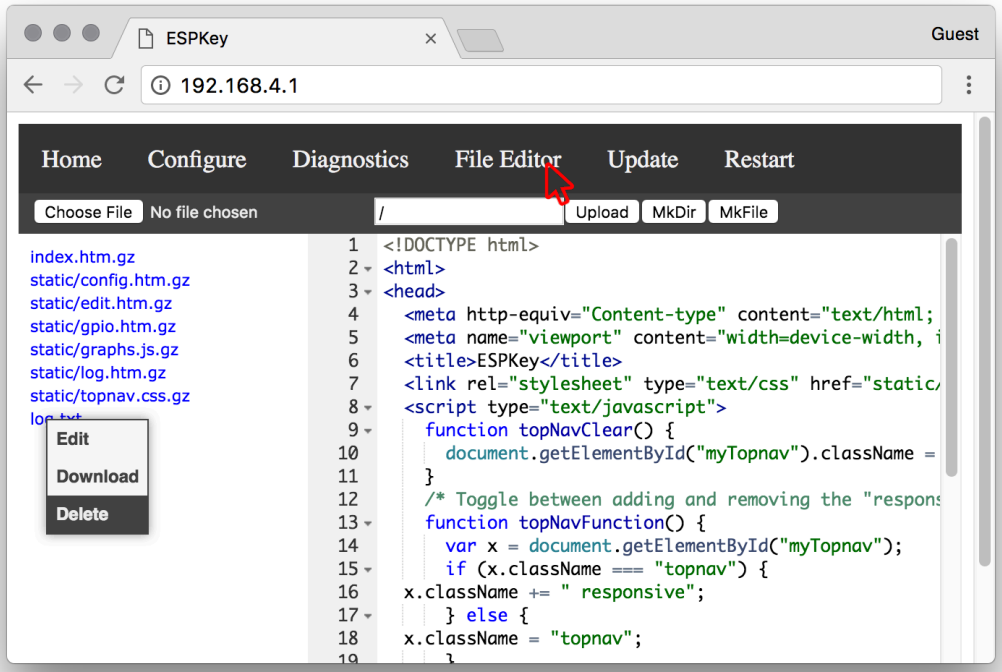
Following activation of DoS mode will remain active until http://<IP or mDNS>/txid?v=6:4 is loaded or the tool is restarted.

Activating DoS via Control Card

DoS may also be activated using a designated “control card”, as defined in the configuration parameters. Once defined, the ESPKey will monitor reader transmissions for bitstream patterns matching that of the defined control card. Once detected, the ESPKey will enter DoS mode and remain so until http://<IP or mDNS>/txid?v=6:4 is loaded or the tool is restarted.

File Editor

The ESPKey UI contains a rudimentary file manager that may be used to manually browse and edit the ESPKey filesystem. Due to the limited memory, available on the ESPKey, the file manager is dynamically loaded from the Internet. If connecting to the ESPKey in AP mode, the file editor will not load unless it has already been cached.



In order to load the file manager reliably, the client device should have a multi-homed network connection that allows the browser to access both local and non-local network resources.

Using File Editor to Restore ESPKey to Factory Defaults

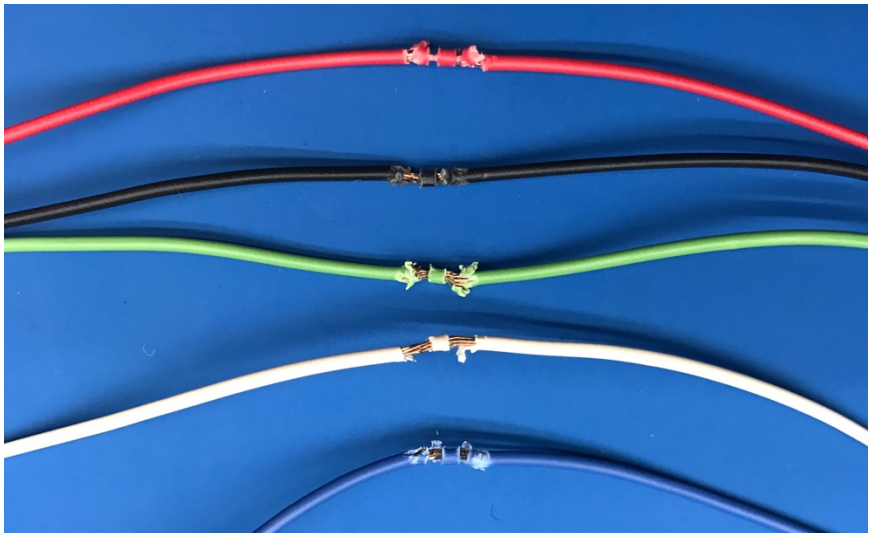
The file editor may be used to manually delete the log.txt and config.json files. The deletion of log.txt will result in the immediate loss of all logged credentials, while the deletion of config.json will result in the ESPKey reverting to factory defaults upon the next power cycle or restart.

Post-Deployment

ESPKey Removal

IDC connections are designed for semi-permanent connections, and are not designed for easy wire removal. Accordingly, precautions must be taken during ESPKey removal to minimize the chance of damage to the key or target wiring.

To remove wires from the IDC, work on each wire one at a time. Grasp each wire as close to the IDC as possible on both sides, and then pull consistently and firmly away from the connector. Care should be taken as improper removal may damage the ESPKey or cause the copper conductor to break.



Following removal, the wire insulation will be damaged. Insulating tape must be applied to the wiring to prevent accidental short-circuits.

The use of insulating electrical tape should not be considered a permanent remedy, and it is strongly advised that reader wiring be cut and re-terminated following the conclusion of the field operator's engagement.

Data Destruction

Following the conclusion of field deployment, it is strongly advised to purge all sensitive data from ESPKey, including saved credential information.

Troubleshooting and Support

Issue:	Log entries only appear to include ‘0’ or ‘f’ characters.
Resolution:	Ensure DoS mode is disabled by restarting ESPKey or transmitting http://<IP or mDNS>/txid?v=6:4
Resolution:	Check the debug page and ensure both green and white lines appear to be “high” on the graph. If either line is “low” check physical IDC connections and try again.
Issue:	Connection to ESPKey Wi-Fi is successful, but UI does not load or is blank.
Resolution:	If dynamic content such as http://<IP or mDNS>/version is served successfully, then re-upload static UI pages via ui-update.sh script.
Resolution:	Re-flash the firmware with the latest version using update instructions.
Issue:	ESPKey Wi-Fi is visible but connection attempts are unsuccessful.
Resolution:	Try connecting from a different device
Issue:	ESPKey Wi-Fi is not visible but power LED’s are illuminated.
Resolution:	Power-cycle the ESPKey
Resolution:	Re-flash firmware via UART (Contact Red Team Tools for Instructions)

Additional Support

For ESPKey devices purchased directly from Red Team Tools, please contact espkey@redteamtools.com with any additional questions. For ESPKey devices purchased elsewhere, please contact the distributing vendor.

Revision History and Changelog

ESPKey Documentation

2017-03-15

Version 1.0.0	Initial public release.

ESPKey Hardware Changelog

XXXX-XX-XX

Version 1.3.1	Pre-production release.

ESPKey Firmware Changelog

2017-01-17

Version 128	Pre-production Release. Fixed recording of credential lengths longer than 40 bits.