SmartWare “Encrypted File Download” HowTo

1. Description

Explains the encrypted configuration download feature of SmartWare.

2. Introduction

TFTP as a configuration download mechanism has the advantage of being extremely simple (trivial) and applicable in any network without any requirements for specialized management servers or applications.

It has the disadvantage of being completely unsecure.

The security hole of downloading complete configurations - which may contain IP addresses, login names and passwords for PPP or VoIP registrations - using TFTP becomes particularly pressing in combination with the auto-provisioning feature which allows large scale distribution of configurations in entire networks.

To alleviate this problem and maintain the simplicity of TFTP downloads support for encrypted configuration file downloads is introduced.

Goal:
Prevent maliciously intercepted configurations to be readable by unauthorized users.

Pre-requisites:
Only authorized users have configuration access to the SmartNode. The configurations can be stored in plain form on the SmartNode.
SNMP Write Access shall be restricted by means of communities and ACLs to prevent unauthorized SNMP initiated configuration downloads.
Telnet access shall be restricted by means of credentials and ACLs.

3. Overview

- An external encryption tool on the PC is used to encrypt the configuration file:
  enctool encrypt <plain-config-file> <enc-config-file> <key>
- The encrypted configuration file can then be downloaded with TFTP triggered by
  - the CLI copy command:
    copy tftp://<host>/<path> <config-file>
  - auto provisioning
  - SNMP
  - HTTP
- On the SmartNode the encryption is detected and the configuration file is automatically decrypted before stored to flash.
- The encryption key can be
  - downloaded to the SmartWare
  - specified with the PC encryption tool
- The encryption key may include the MAC address and/or serial number of the SmartNode using the placeholders $(system.mac) and $(system.serial) respectively.
- An encrypted configuration file can be uploaded to a TFTP server on request, specifying the encrypted flag:
  copy <config-file> tftp://<host>/<path> encrypted
- On the PC the encryption tool can be used to decrypt the file:
  enctool decrypt <enc-config-file> <plain-config-file> <key>
- A log file lists the last up/downloads:
  show log file-transfer

4. Use Cases

Install the encryption key

You must install an encryption key with the SmartNode. The encryption key is used to automatically decrypt an encrypted configuration file that is downloaded later.

To install the encryption key, you have to create a file on your TFTP server that contains the key. Then you have to download this key file to the SmartNode using the 'copy' command of the SmartNode:

The key file shall contain a key string of at most 24 characters on a single line. Spaces, tabs and LF/CR characters are trimmed. The key must not contain LF/CR, spaces, tabs, quotes (“),backslashes (\) or the null character. If the key contains more than 24 characters, only the first 24 characters are considered. The key may contain variables that are resolved when the key file is downloaded to a SmartNode. Using this mechanism you can specify device-specific encryption keys. We currently support the following variables:

- **$(system.mac)**: The MAC address of the first ethernet port. Execute the **show port ethernet** command on a SmartNode to display the MAC address of a SmartNode. This value without the colon separators and with all lower-case hexadecimal letters is used instad of the variable on the SmartNode.
- **$(system.serial)**: The serial number of the SmartNode. Execte the **show verion** command on the SmartNode to display the serial number. Unlike the MAC address, *don't* convert the serial number to lower-case letters.

When your key file contains the following line…

123**$(system.serial)**abc**$(system.mac)**XYZ

show port ethernet shows the following…

```
Ethernet Configuration
------------------------------------
Port            : ethernet 0 0 0
State           : OPENED
MAC Address     : 00:0C:F1:87:D9:09
Speed           : 10MBit/s
Duplex          : Half
```

```
Encapsulation   : ip
Binding         : interface eth0 router
```

and **show version** the following….

```
Productname       : SN1200
Software Version  : R3.20 TB2005-06-24_MEYER SIP
Supplier          :
Provider          :
Subscriber        :

Information for Slot 0:
SN1200
Hardware Version  : 0004, 0001
Serial number     : 000CF187D909
Software Version  : R3.20 TB2005-06-24_MEYER SIP
```

the encryption key on this SmartNode will be interpreted as…

123**000CF187D909**abc**000cf187d909**XYZ

Then you have to download the created key file to the SmartNode. Open a telnet session and type in the following commands:

>**enable**
#copy tftp://<ip>/<path> key:

where <ip> is the IP address of your TFTP server and <path> is the path to the key file relative to the TFTP root.

Important Note: The downloaded key also defines how the passwords are encrypted in your configuration files. After you downloaded a key file you have to regenerate the startup-config from the running-config by executing the command

copy running-config startup-config

If you don't do this the device will fail executing the commands that have encrypted password arguments, e.g. 'administrator', 'h235-security password', etc.

Encrypt a configuration file

Use the encryption tool to encrypt a configuration file on your PC. Therefore you have to enter the following command.

enctool encrypt <plain-file> <encrypted-file> <key>

where <plain-file> is the path of the non-encrypted input configuration file and <encrypted-file> is the path of the encrypted output configuration file. <key> specifies the encryption key which shall be used to encrypt the configuration file.

Download an encrypted configuration file

Now you can download the configuration file as usual using the CLI copy-command, the auto-provisioning feature, HTTP or SNMP download. The SmartNode automatically detects that a downloaded file is encrypted and tries to decrypt the file using the installed key.

Upload an encrypted configuration file

The SmartNode immediately decrypts a configuration file after downloading it. This is the configuration file is stored non-encrypted in the flash memory. Thus when you upload a configuration it is uploaded non-encrypted.
You may upload an encrypted configuration file specifying the **encrypted** flag at the end of the copy command:

#copy startup-config tftp://<ip>/<path> encrpted

This encrypts the configuration file before sending it to the TFTP server. Use the **enctool decrypt** command on the PC to regain the original configuration.

File Transfer Logs

We introduced an additional log file that stores the history of all file transfers (up to 50 entries). To show all recently executed file transfer operations enter the following command:

#show log file-transfer