Firmware signieren

Inhalt

Über diese Seite

Hier wird das signieren unserer Firmware beschrieben. Es ist also eine Anleitung für die Administratoren und wird für die Nutzung von Freifunk selbst nicht benötigt.

Hintergrundwissen

- Konfiguration beim Freifunk Münsterland für good_signatures im stable: 2 (=Kein Admin kann alleine handeln)
- Die Platzhalten <secretDatei> muss im Script durch den Dateinamen mit dem Secret-Schlüssel ersetzt werden.
  Niemals das Secret selbst im Klartext eingeben um einen leak per bash-history etc. zu verhindern.
- Den Platzhalter <public> im Script muss durch den persönliche Public-Schlüssel ersetzt werden.
- Das Script muss als root ausgeführt werden. Ursache unbekannt. Als User hat es bei mir nicht funktioniert.

Voraussetzungen

- Auf dem lokalen Rechner muss das Tool 'ecdsautil' (https://github.com/freifunk-gluon/ecdsautils) installiert werden.
- Auf dem lokalen Rechner muss der geheime Schlüssel als Datei gespeichert sein (Anleitung zum Erstellen: https://github.com/freifunk-gluon/ecdsautils#generate-key)
- Der geheime Schlüssel muss zum signieren berechtigt sein (=In der Konfiguration der Knoten gelistet sein https://github.com/FreiFunkMuenster/site-ffmsl)
- Benötigtes Script: https://raw.githubusercontent.com/freifunk-gluon/gluon/master/contrib/sign.sh
- SSH-Zugriffsberechtigung auf den Firmware-Server
- Zu nutzender SSH-Key muss bereits geladen sein (ssh-add <DateinameZumSchlüssel>)
- Ein persönlicher Schlüssel (secret+public) für die Signierung muss bereits erstellt worden sein und in der Konfiguration der Knoten berechtigt sein.

Ablauf

Vorbereitung (Einmalig)

ECDSAutils bereitstellen

- Tools installieren
  - sudo apt-get install cmake git make pkg-config sshfs
- libuecc kompilieren / installieren
  - git clone http://git.universe-factory.net/libuecc
  - cd libuecc
  - cmake ./
  - make
  - sudo make install
  - sudo ldconfig
- ECDSutil kompilieren / installieren
  - git clone https://github.com/freifunk-gluon/ecdsautils.git
  - cd ecdsautils
  - mkdir build
  - cd build/
  - cmake ../
  - make
  - sudo make install

    - sudo ldconfig
- Scripte herunterladen in /usr/bin
    - wget https://raw.githubusercontent.com/freifunk-gluon/gluon/master/contrib/sign.sh
    - wget https://raw.githubusercontent.com/freifunk-gluon/gluon/master/contrib/sigtest.sh
- Ordner zum mounten der Manifest-Dateien erstellen
    - mkdir remotefiles

Vorbereitung (jedes Mal)

- Prüfen ob die Änderungen für das aktuelle Release verständlich / nachvollziehbar sind. Neue Versionen nicht ungeprüft signieren. Siehe auch "Regelwerk" der Adminvereinbarung.

Durchführung

- Mounten des Ordners auf dem Firmware-Server
    - `sshfs -p 223 root@firmware.ffmsl.de:/var/www/html/ remotefiles`
- Signieren der einzelnen Dateien
    - `sign.sh <secretDatei> remotefiles/domaene07/versions/v4.1.0/sysupgrade/stable.manifest`

      `Hier natürlich den Pfad auf die Manifest-Datei ändern, die signiert werden soll`
- Testen ob richtig signiert wurde
    - `sigtest.sh <public> remotefiles/domaene07/versions/v4.1.0/sysupgrade/stable.manifest`
    - `echo $?`

      `--> 0 bedeutet gültig`

      `--> Nicht 0 bedeutet ungültig`

Nacharbeiten

- Den Remote-Ordner wieder entmounten
    - `fusermount -u remotefiles`

Tipp

Das Script https://raw.githubusercontent.com/FreiFunkMuenster/tools/master/signieren.sh automatisiert die Signierung nach diesem Prinzip für mehrere Domänen/Branches/Versionen gleichzeitig