Tool Software

Cyber Security Precautions for Secondary Development

Issue 00B01

Date 2018-03-20

HiSilicon (Shanghai) Technologies Co., Ltd.

Address: New R&D Center, 49 Wuhe Road, Bantian,
Longgang District,
Shenzhen 518129 P. R. China

Website: http://www.hisilicon.com/en/

Email: support@hisilicon.com

About This Document

Purpose

This document describes the disclaimer for the tool software.

Product Version

The following table lists the product versions related to this document.

Product Name	Version
Hi3559A	V100
Hi3559C	V100
Hi3519A	V100
Hi3516C	V300
Hi3516E	V100
Hi3556A	V100
Hi3516C	V500
Hi3516D	V300
Hi3559	V200
Hi3556	V200
Hi3516E	V200
Hi3516E	V300
Hi3518E	V300
Hi3516D	V200

Intended Audience

This document is intended for the target customers of the tool software solutions.

Change History

Changes between document issues are cumulative. The latest document issue contains all the changes made in earlier issues.

Issue 00B01 (2018-03-20)

This issue is the first draft release.

Contents

Figures

Tables

1 Overview

The tool software package of the terminal chip is developed to meet product requirements. As part of the continuous accumulation of the delivery capabilities of HiSilicon tools, the tool software package provides overall tool solutions for both internal and external customers, improving O&M development efficiency.

The tools in this package are external debugging and assessment tools, used only for customers during chip secondary development to improve efficiency, enhance service quality, and reduce risks. You are advised to use the tools in a non-commercial environment to avoid network security risks.

1.1 Description of Version Delivery

The basic version is developed in C/C++ and Java. Table 1-1 lists the tools and products delivered with the version.

Table 1-1 Tools and products delivered with the version

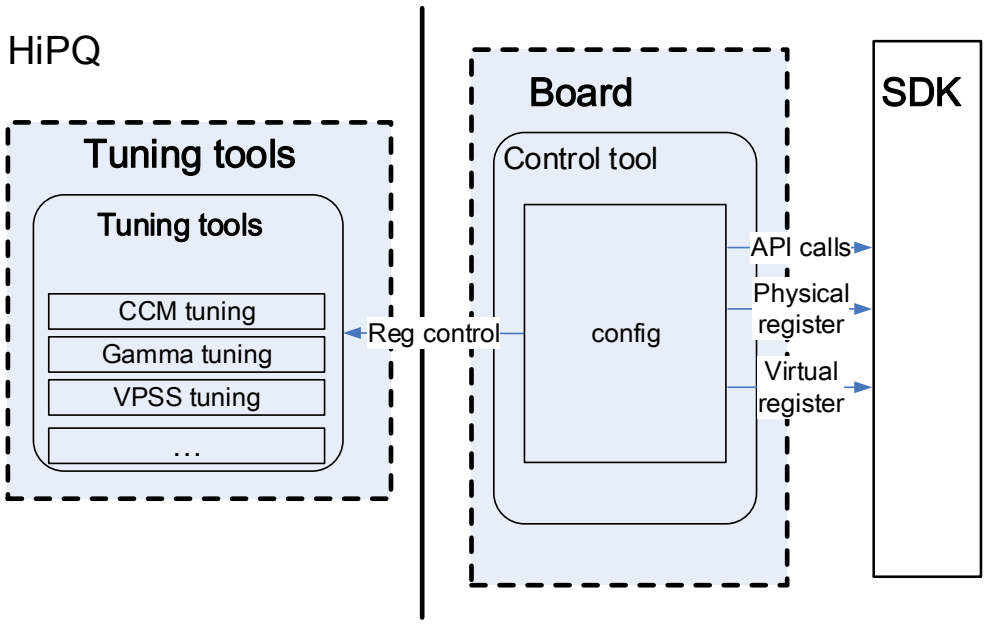
Tool Name	Tool Type	Purpose
HiBurn	External debugging and assessment tool	Mass production and image burning
HiLoader	External debugging and assessment tool	Image creating
HiAQ	External debugging and assessment tool	Audio quality debugging
HiPQ	External debugging and assessment tool	Picture quality debugging
CASignTool (Linux/Windows)	External debugging and assessment tool	Security protection

2 HiPQ

2.1 Architecture

Figure 2-1 shows the HiPQ architecture.

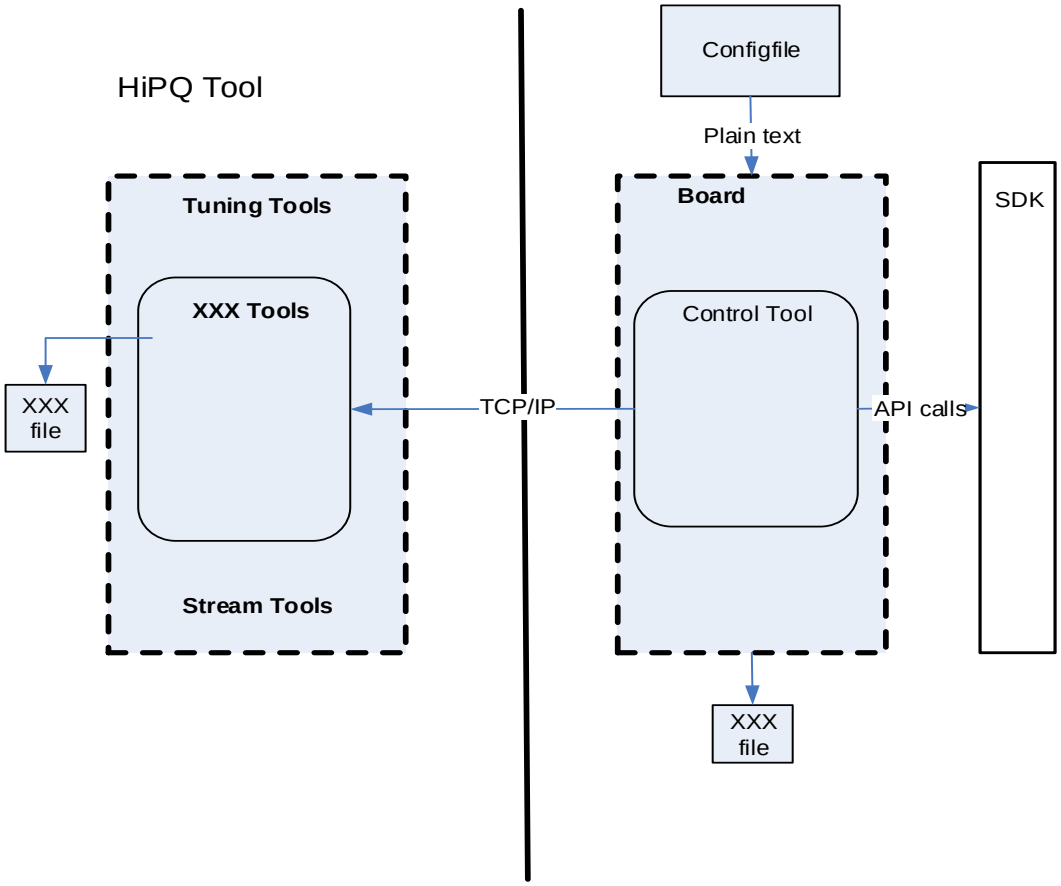
Figure 2-1 HiPQ architecture



2.2 Security Attacks and Threats

Figure 2-2 shows the interaction between HiPQ and the peripherals.

Figure 2-2 Diagram of the interaction between HiPQ and the peripherals



HiPQ consists of an executable program at the server end and an executable program at the client end. Data is transmitted between the two ends over LAN TCP/IP, and HTTP. During network construction:

- For the online tuning part, the server reads the port number from the plaintext configuration files and then monitors ports. The server stores binary files (.bin file and .raw file) during running.
- For the online tuning part, the client specifies the TX port ID and then implement command and data exchanges. The client stores data (raw data, YUV data, and ISP parameters) obtained from the server during running.
- For the online on-demand part, the server reads some default media parameters from the plaintext configuration files and specifies the port 80 fixedly as the listening port. The client stores the recording files (.h264 and .h265 files) and snapshot files (.jpg files) during running.

2.3 Security Dimensions

2.3.1 Login Control

- HiPQ runs on the MPP environment, including board preparation, boot burning, kernel burning, rootfs burning, and network setting.

- Before you run HiPQ, you need to configure the monitoring port ID for the tuning part. You can find the port ID in **config.cfg**, which is saved in the HiPQ directory in plain text mode. Then, change the value of the monitoring port parameter **[Default]**. The default monitoring port ID is **4321**.

  For details about the modification, see section 1.3.2.3 in the *HiSilicon PQ Tools User Guide*. The VOD part uses port 80 by default. Before running HiPQ, check whether other processes occupy port 80.
- To log in to the HiPQ client, you need to enter the IP address and port ID of the board to be logged in to. The port ID for online on-demand is **80**, which does not need to be entered during login.

2.3.2 Permission Control

When logging in to the HiPQ client, you need to enter the IP address of the board to be logged in and port number (You can skip the port number for the VOD part because it uses port 80 by default.)

2.3.3 Storage Security

The intermediate files are saved to and read from the board or PC according to different application conditions during the running of HiPQ. You can configure a directory for storing the server intermediate files. Configuration items are all saved in plaintext to the **config.cfg** file in the running directory of HiPQ. You need to select the directory for reading and saving the client intermediate files such as .yuv and .raw files. While data reports, in formats such as .jpg, are stored in the default directory of the client.

2.3.4 Interaction Security

HiPQ runs over the LAN only. It uses LAN TCP/IP to implement data exchanges.

2.3.5 Data Transmission Security

During data transmission, HiPQ uses simple flags to identify whether a packet is sent by HiPQ. The packets integrity is verified with simple algorithms. Only preliminary identification is performed to determine whether packet loss occurs on the network.

2.4 Security Domains

2.4.1 Management

HiPQ is designed for product development only. You are not advised to integrate it in a product.

2.4.2 Control

- HiPQ is designed for product development only.
- The involved port needs to be disabled or deleted in the product. You need to protect the port ID and the board IP address.
- The access to intermediate sensitive information, such as the binary function, should not be included in the product. Such functions need to be removed from the product.

2.4.3 Environment

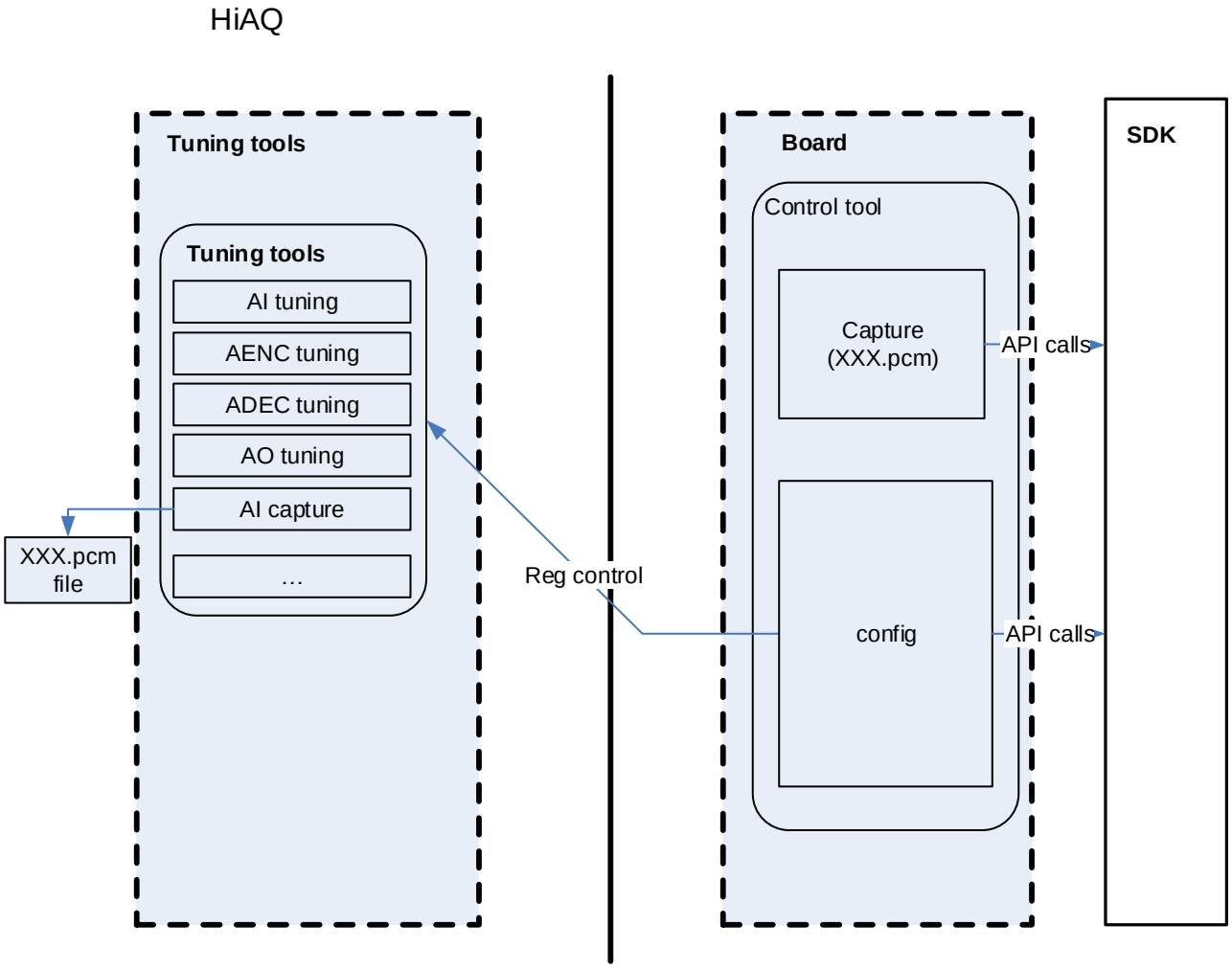
HiPQ can only be used over the LAN.

3 HiAQ

3.1 Architecture

Figure 3-1 shows the HiAQ architecture.

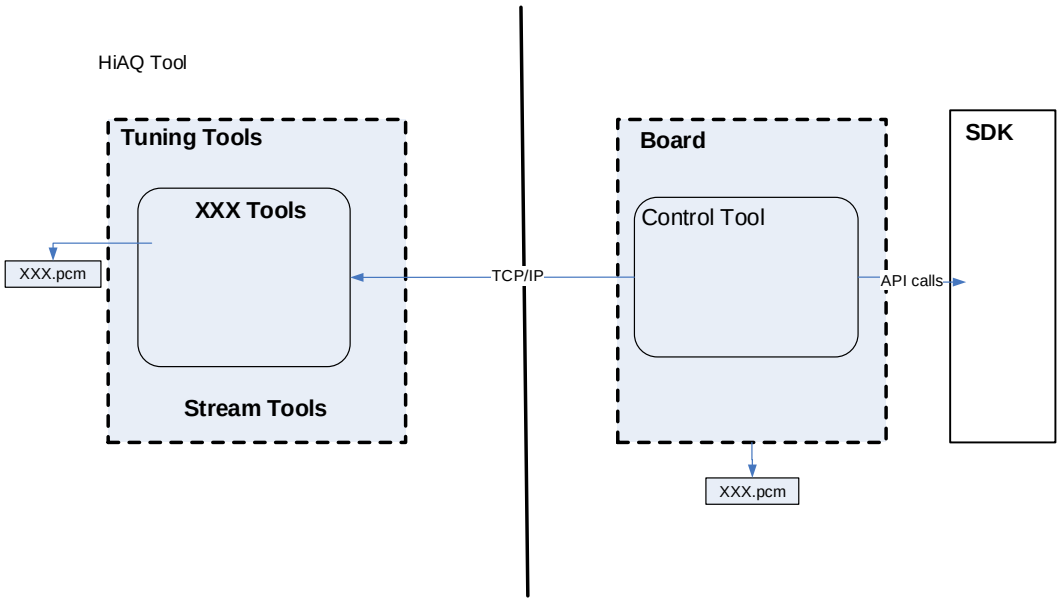
Figure 3-1 HiAQ architecture



3.2 Security Attacks and Threats

Figure 3-2 shows the interaction between HiAQ and the peripherals.

Figure 3-2 Diagram of the interaction between HiAQ and the peripherals



HiPQ consists of an executable program at the server end and an executable program at the client end. Data is transmitted between the two ends over LAN TCP/IP, and HTTP. During network construction:

- For the online tuning part, the server reads the port number from the plaintext configuration files and then monitors ports. The server stores binary files (.pcm files) during running.
- For the online tuning part, the client specifies the TX port ID and then implements command and data exchanges. The client stores data (.pcm data) obtained from the server during running.

3.3 Security Dimensions

3.3.1 Login Control

HiPQ runs on the MPP environment, including board preparation, boot burning, kernel burning, rootfs burning, and network setting. Before running HiPQ, you need to configure the listening port number for the tuning part, which is an input parameter of the **HiAQTool.sh** file. Specifically, run **./HiAQTool.sh** *port number*.

3.3.2 Permission Control

When logging in to the HiPQ client, you need to enter the IP address of the board to be logged in and port number

3.3.3 Storage Security

The intermediate files are saved to and read from the board or PC according to different application conditions during the running of HiPQ. Server intermediate files are read from or saved to the running directory of HiPQ. You need to select the directory for reading and saving the client intermediate files such as .pcm files.

3.3.4 Interaction Security

HiPQ runs over the LAN only. It uses LAN TCP/IP to implement data exchanges.

3.3.5 Data Transmission Security

During data transmission, HiPQ uses simple flags to identify whether a packet is sent by HiPQ. The packets integrity is verified with simple algorithms. Only preliminary identification is performed to determine whether packet loss occurs on the network.

3.4 Security Domains

3.4.1 Management

HiPQ is designed for product development only. You are not advised to integrate it in a product.

3.4.2 Control

HiPQ is used during product development only, and users need to protect the port ID, board IP address, and sensitive intermediate information.

3.4.3 Environment

HiPQ can only be used over the LAN.

4 CASignTool

4.1 Security Dimensions

CASignTool provides the following three encryption methods and modes for the encryption/decryption module (Crypto) and non-boot signature module.

Encryption methods:

- AES
- TDES
- SM4

Encryption modes:

- CBC
- ECB
- CTR (different from that in the Chinese national encryption algorithms)

You are advised to use AES+CBC, which provides the best security result.

The Crypto module supports Chinese national encryption algorithms (SM3 and SM4 encryption algorithms, and SM2 signature algorithm), which are disabled by default for markets outside China.

4.2 Security Domains

4.2.1 Control

The tool is used during product development only, and users need to protect the port ID, board IP address, and sensitive intermediate information.

4.2.2 Usage

The delivered tools do not involve network connection, nor obtains customer privacy data. The final product does not include the delivered tools.

5 Tools for Mass Production, Image Burning, and Other Purposes

5.1 Security Attacks and Threats

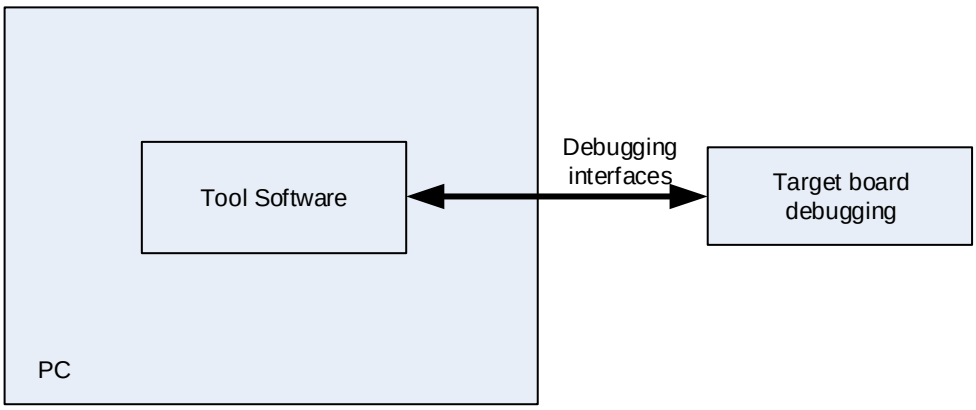
The tools are the standalone software running on the PC. Some tools are associated only with the file systems running on the system. Some debugging tools or burning tools need to communicate with the target board through debugging interfaces. The involved interfaces are shown in Table 5-1.

Table 5-1 Tools and communication modes

Tool	Communication Mode
Hiburn	Serial port, USB interface, I^2C interface, JTAG interface, and network port

Figure 5-1 shows the networking structure of the debugging tools and image burning tools.

Figure 5-1 Networking structure of the debugging tools and image burning tools



As described in Table 5-1, the target board is debugged through a number of debugging interfaces, such as the JTAG interface, serial port, I^2C interface, and network port. You can print debugging information through the debugging interface. You can even access the

internal registers of the CPU and any information in the device. Note that these are important means for hackers to analyze the hardware, software, and configuration parameters of a device.

For example, using the JTAG interface, hackers can trace, commission, and execute arbitrary code, read and change the software and sensitive configuration data stored in the flash memory, and locate the security vulnerabilities for attacking. If the hacker publishes the security vulnerabilities and attacking methods on the Internet, substantial security threats will be posted to your product.

The R&D board allows debugging interfaces to be reserved for debugging only. However, the design of the debugging interface needs to be removed for the official delivery. The interface sockets should be removed as well.

5.2 Security Domains

5.2.1 Management

The tools are used only during product development and cannot be used in the product. Otherwise, HiSilicon is not responsible for any security problem.

5.2.2 Control

The tools are used during product development only, and users need to protect the port ID, board IP address, and sensitive intermediate information.

5.2.3 Usage

The delivered tools do not involve network connection, nor obtains customer privacy data. The final product does not include the delivered tools.

5.3 Other Security Precautions

5.3.1 JTAG Interface

Attackers may modify the system and configurations through the JTAG interface to damage the system.

You are advised to take the following measures:

- Remove the JTAG interface physically from the products to be delivered.
- The chip provides the JTAG disable function, which allows the permanent disabling of the JTAG interface from the chip level.

5.3.2 PC Debugging Tool

- All internal tools are only for R&D employees in the device chipset department and cannot be contained in the official release package of the product.
- All R&D debugging tools can only be used for function or effect tuning during product development or production. Do not include them in products to be delivered.

- All service O&M tools are provided only for specified partners and cannot be publicly accessed.

5.3.3 Debugging Interface

The R&D board allows debugging interfaces to be reserved for debugging only. However, the design of the debugging interface needs to be removed for the official delivery. The interface socket should be removed as least.

5.3.4 Image Security

The images provided for debugging cannot be compiled to the products to be delivered.

6 Conclusion

Based on the preceding chapters, the network security precautions are summarized as follows:

- The tools provided in the tool software package are external debugging and assessment tools or production tools, which are used only during the development process.
- The product developer is responsible for the protection of the product data.
- The tool software package contains executable files and configuration files. In the software package related to mass production, delete all executable files and configuration files of the tools and delete the related description from the document.
- All tools provided are non-commercial tools. You can use them only for product development. Otherwise, HiSilicon is not responsible for any security problem.

A Acronyms and Abbreviations

Table A-1 Acronyms and Abbreviations

Acronyms or Abbreviations	Full Name
AES	advanced encryption standard
BLPK	boot loader public key
CBC	cipher block chaining
CTR	counter
ECB	electronic codebook
HTTP	HyperText Transfer Protocol
I^2C	inter-integrated circuit
ISP	image signal processor
JTAG	joint test action group
LAN	local area network
MPP	media process platform
SDK	software development kit
TCP	Transmission Control Protocol
TDES	triple data encryption algorithm
USB	universal serial bus

