

IPENCRYPTER.COM

HDL Intellectual Property Protection

Encryption, License Management and Rights Evaluation

based on
IEEE Std 1735™-2014 standard

ipencrypter.com

Contents

- Introduction 2
- Features of *ipencrypt* 2
 - Encrypt IP 3
 - Process of creating encrypted IP through *ipecrypt*:..... 3
 - Decrypt IP within EDA Tool 4
 - Client Module..... 4
 - License Management 5
 - Rights Evaluation..... 5
 - License Proxy..... 5

Introduction

The *ipencrypter* is a suite that provides modules for encryption, license management and rights evaluation for intellectual property (IP) conforming to IEEE Std 1735™-2014 standard (a.k.a. P1735 v2). This standard specifies syntax for HDL IP encryption, rights and license management which is embedded in HDL. IP author can protect IP source code and its usage by inserting special “protect” directives and encapsulating the block of code. *ipencrypter* supports SystemVerilog and VHDL languages.

Features of *ipencrypter*

ipencrypter provides following features:

- Encrypt IP (*ipecrypt*)
- Decrypt IP (include Rights evaluation and License management) within EDA tool (*ipeclient*)
- License Proxy (*ipeproxy*)

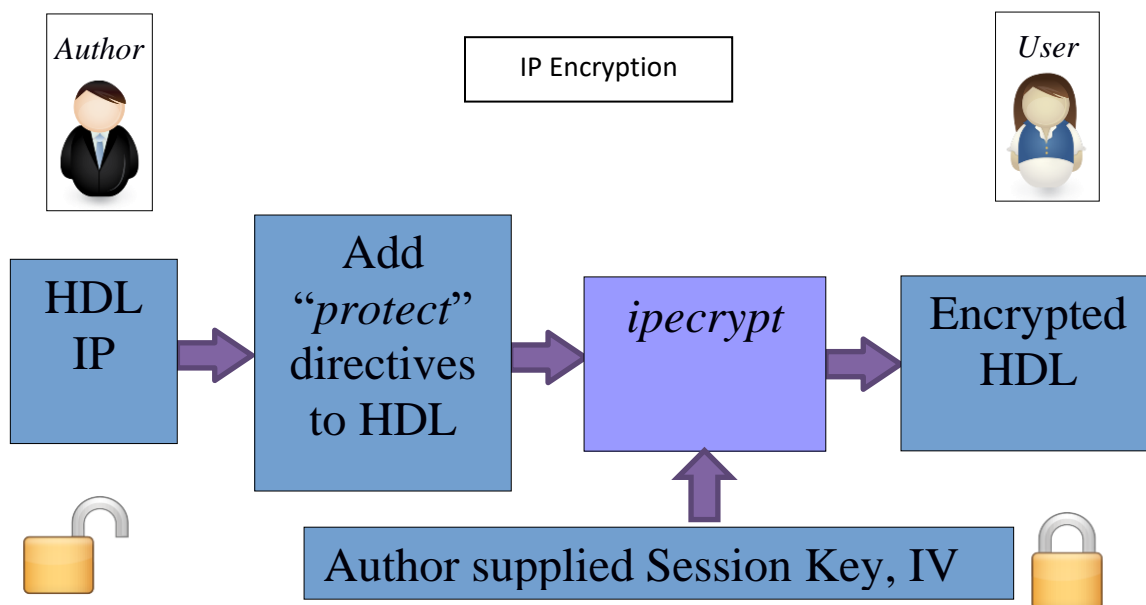
Encrypt IP

IP author can use *ipecrypt* to encrypt an IP. IP author provides the level of protections through protect directives in common and tool blocks. IP author can choose the tools to support and needs public key for each tool.

Process of creating encrypted IP through *ipecrypt*:

- Create IP in plain text
- Add protect directives around the code to encrypt or alternatively specify directives through a separate file
- Run *ipecrypt* application to generate encrypted IP

Encryption Flow:



ipecrypt command syntax for encryption:

- Arguments:
 - --help produce help message
 - -I [--infile] arg input file to encrypt
 - -O [--outfile] arg output file encrypted
 - -F [--force] force overwrite
 - -K [--sessionkey] arg base64 encoded session key for encryption, if empty system generated key will be used
 - --iv arg base64 encoded initialization vector for encryption, if empty system generated key will be used
 - -D [--directive] arg protect directives, if no directive is found in HDL, the directives from this file will be used

- -V [--vhdl] whether its VHDL file (default is verilog)
- --verbose arg verbosity level [0-4] (fatal, error, warn, info or debug)

Decrypt IP within EDA Tool

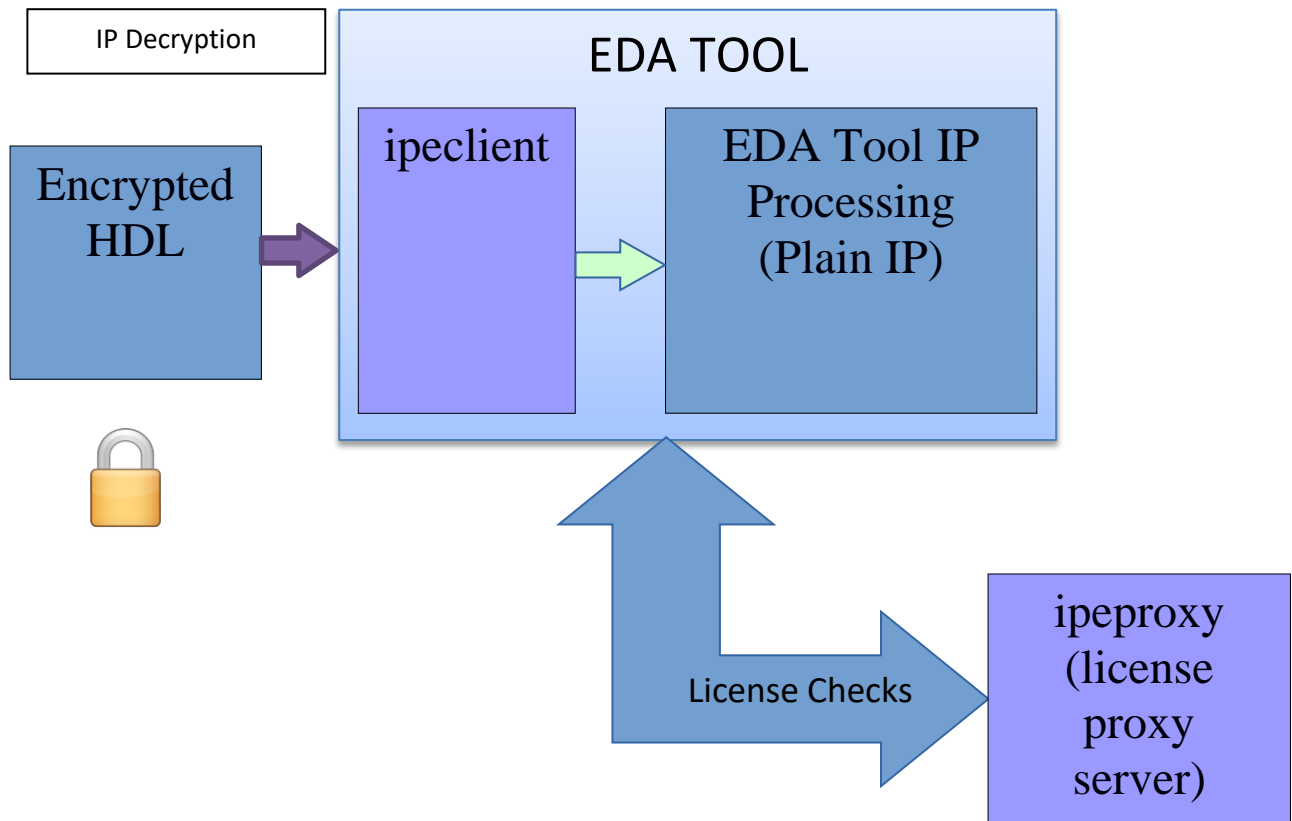
ipencrypter provides a client module, *ipeclient*, that can be used to decrypt IP. It reads the encrypted IP and decrypts it using the tool's private key. The use of IP may further require runtime license check. This module is integrated with the EDA tool.

Client Module

ipeclient is linked to EDA tool. Client parses the IP and performs integrity check. Then it checks for licensing requirement for decryption. If needed it locates the license proxy, establishes secure communication with it and evaluates license rights. Based on the rights granted the tool will either proceed with processing IP or terminates the processing of the IP.

Decryption Flow:

- Tool reads the encrypted IP through *ipeclient* module embedded in it
- *ipeclient* decrypts the IP
- Check license rights for decryption
- Provides IP in plain text to tool
- Provides methods to evaluate rights



License Management

IP author can grant access of IP to users of a particular tool through rights specified in common and tool blocks. The access can further be controlled through licensing mechanism. IP author can distribute license for IP to some users of the tool. In this way IP author can further restrict the use of IP. A license can grant different rights to different users.

Rights Evaluation

The tool vendor requires protecting all aspects of IP. The rights management pragmas provide ability to IP author on how IP may be used. *ipeclient* evaluates complex *control* expressions to determine available rights. The *ipeclient* module provides APIs for this purpose.

License Proxy

ipeproxy is an executable which serves license requests of *ipeclient*. It securely communicates with *ipeclient* and grants or deny license. *ipeproxy* provides APIs to link and get license information from third party license managers. IP author provides the license proxy to IP users.

ipeproxy command syntax:

- Arguments:
 - --help produce help message
 - --port arg (=27500) proxy port number
 - --verbose arg verbosity level

License Proxy-Client Communication:

