

Padlock 2.0 Command Line Guide



The latest version of this document is always available on the Padlock Support Wiki at [http://support.padlocksoftware.net/display/padlock/Padlock 2.0 Command Line Guide](http://support.padlocksoftware.net/display/padlock/Padlock+2.0+Command+Line+Guide)

Padlock ships with several command line tools for creation and validation of licenses. This short guide discusses the tools and gives a few short demonstrations on usage.

Command Line Tools

The following command line tools ship with Padlock:

- **KeyMaker** - Used to generate the DSA KeyPairs required to sign and validate licenses. Keys can also be created from the API directly or by the Padlock Manager application. See the Integration Guide for more details on usage.
- **LicenseMaker** - Creates signed licenses using a previously generated KeyPair and any number of custom parameters for specifying start and expiration dates as well as any number of license properties.
- **LicenseValidator** - Used for getting the state of a license file. This takes a KeyPair and License file as parameters, and will run the License through a series of validation tests to determine the license state. See the Integration Guide for a complete walkthrough of what validation entails.

The above tools make use of the public Padlock API. Anything these tools can do can also be implemented by any third party programmer in a matter of a few hours. See the Padlock JavaDocs (shipped with the Padlock distribution) for complete API documentation.

KeyMaker

The KeyMaker is responsible for generating DSA keys utilized by Padlock. Its usage has been simplified from Padlock 1.x, since all DSA keys are hard coded at 1024 bits.

Usage: `java -jar keymaker.jar <Output key file>`

Example: `java -jar keymaker.jar test.key`

This will product a key file which can be viewed in any text editor. Here is a usage sample:

```
PS D:\Padlock-2.0> java -jar keymaker.jar test.key
Your public key code:
    private static final String publicKey =
        "308201b83082012c06072a8648ce3804013082011f02818100fd7f53811d" +
        "75122952df4a9c2eece4e7f611b7523cef4400c31e3f80b6512669455d40" +
        "2251fb593d8d58fabfc5f5ba30f6cb9b556cd7813b801d346ff26660b76b" +
        "9950a5a49f9fe8047b1022c24fbba9d7feb7c61bf83b57e7c6a8a6150f04" +
        "fb83f6d3c51ec3023554135a169132f675f3ae2b61d72aef22203199dd1" +
        "4801c70215009760508f15230bccb292b982a2eb840bf0581cf502818100" +
        "f7e1a085d69b3ddecbbcab5c36b857b97994afbbfa3aea82f9574c0b3d07" +
        "82675159578ebad4594fe67107108180b449167123e84c281613b7cf0932" +
        "8cc8a6e13c167a8b547c8d28e0a3ae1e2bb3a675916ea37f0bfa213562f1" +
        "fb627a01243bcca4f1bea8519089a883dfe15ae59f06928b665e807b5525" +
        "64014c3bfecf492a0381850002818100f7746f79a712ae0865ddfa585fb0" +
        "4a990f7933aa73124dab89e4af80b2a9a0ab57dac23995326d6fd7db2950" +
        "c1260edfeb7dffe9f4e3afeda65c77c2565a58127f922b571ebf70091d5" +
        "c8356437e0481172682e2f817cd4cade78d0c3734a13b7a735a11e2b0b55" +
        "0f20370f2e315b6050cc0a6f3682cf6ac15d556a007b10fd" ;
PS D:\Padlock-2.0>
```

LicenseMaker

Basic Usage

The LicenseMaker generates signed license files based on a number of optional parameters and two required parameters: The license output file and KeyPair file. The simplest usage is the following:

Usage: `java -jar licensemaker.jar -o <Output License File> -k <Keypair File>`

Example: `java -jar licensemaker.jar -o test.lic -k test.key`

The above generates a nearly empty license file (test.lic). If you were to view its contents, the only items you'll see are a license creation date, start date, signature, and version. If you're using Padlock without a valid license, you'll also see an expiration date for two weeks after the license start date.

Sample license file output:

```
#Wed Mar 03 19:22:26 EST 2010
startDate=1267662146986
signature=302c02145440aa22ec94d210ccbfe8e98a131155ca02f57c021422598935dea797ff055a11f7bbca6c9d377735c
version=2
creationDate=1267662146986
```

While the above license will validate, it's not terribly useful without some properties attached to it. See below for an example of how to add properties.

Setting dates

Let's set a license start date that differs from the current date. There are two ways of setting the start date: either with the -s flag and passing the number of milliseconds since the epoch, or using the -S flag and passing a human readable date and optional date format (as specified by Java's SimpleDateFormat). Let's look at the latter.

Usage: `java -jar licensemaker.jar -o <Output License File> -k <Keypair File> -S <Start Date> [Date Format]`

Example: `java -jar licensemaker.jar -o test.lic -k test.key -S 01/02/1999 MM/dd/yyyy`

The above license became valid on January 2, 1999. If you go with the default date format yyyy/MM/dd, you can skip entering the format in:

Example: `java -jar licensemaker.jar -o test.lic -k test.key -S 1999/01/02`

The same format rules hold true for the expiration date. Just remember that if you're using a demo version of Padlock, it will overwrite your expiration date with one set to two weeks after the start date.

Floating Expiration Date

Next up let's look at the Expiration Float property. This will set a floating expiration date for a license. This parameter is how you would create something like a 14 or 30 day trial license. The clock on the floating expiration starts from the time the license is first run through the validator, which in most cases would be during the program's first use.

Usage: `java -jar licensemaker.jar -o test.lic -k test.key -x <milliseconds to float for>`

Example: `java -jar licensemaker.jar -o test.lic -k test.key -x 3600000`

The above would let set a license validity of one hour after the first validation attempt. Note: The LicenseValidator command line tool (discussed below) does not trigger the float flag, and will not apply the results of this property towards validation. In short, at this point you've got to write some test code for yourself to test this property out.

License Properties

A license can have any number of properties, which are developer defined key/value pairs. This is the place to store items such as end user name, emails, support contract lengths, software versions, etc. To add properties to a license simply use the -p flag as such:

Usage: `java -jar licensemaker.jar -o <Output License File> -k <Keypair File> -p <"property1=value1, property2=value2">`

Example: `java -jar licensemaker.jar -o test.lic -k test.key -p "name=John Smith, email=john@smithco.com"`

Here's the contents of the license file:

```
#Wed Mar 03 19:22:26 EST 2010
property_name=John Smith
startDate=1267662146986
signature=302c02142b627cb4eb52283895a67971ebf9af2139ef3980021436b23a62f9eec35a3f1ed0091f9a0a8e5f3d0b07
version=2
creationDate=1267662146986
property_email=john@smithco.com
```

The contents of a license file are never in any particular order (just the whim of the Java Properties instance that exported them). All user defined properties start with the `property_` prefix.

Hardware Locking

Lastly, a license can be limited on what computer systems it runs on by means of hardware locking the license to MAC addresses. This is a new license feature of Padlock 2.0, and is done with the `-h` flag.

Usage: `java -jar licensemaker.jar -o <Output License File> -k <KeyPair File> -h <comma separated MAC addresses>`

Example: `java -jar licensemaker.jar -o test.lic -k test.key -h "00:25:00:3e:8c:45, 00:25:00:3e:8c:46"`

The above command will bind the license to systems having at least one of the above MAC addresses. Note: Java on OS X has a bug where interfaces that aren't active cannot be discovered, and may cause false validator failures.

License Validator

The License Validator is a useful tool for testing the validity of signed licenses. When run with a KeyPair file and a License file as parameters, it will attempt to verify the signature of the license, check that the current date is within the start and expiration dates, print out the license properties, and then check for hardware locking.

Basic Usage

The only two parameters permitted by the License Validator is the License and KeyPair files. Usage is as follows:

Usage: `java -jar licensevalidator.jar -l <License File> -k <KeyPair File>`

Example: `java -jar licensevalidator.jar -l test.lic -k test.key`

Conclusion

This completes the overview of the command line tools in Padlock 2.0. If you have any questions or issues, please email support@padlocksoftware.net.

