CD Key

General Concept

A license key for a software is some sort of alphanumeric series of characters, that allow for a piece of software to be used under a certain license. In the gaming sector, license keys are very often referred to only as keys or CD keys. The concept of license keys is a general mechanism of a kind of authentication/prove, that a piece of software was actually obtained by buying a license, instead of just copying an installer or an installation medium (e.g. a CD). As these license keys for games were very often provided along with the installation CD, the term CD key was coined. The installation was only finished (or even started) when providing a valid key to the installer.

Game Keys on Online Platforms

In times of platforms like steam or gog, keys returned to their original function, to prove a genuine license was obtained, while no CDs or other installation media are actually shipped any more. The game is only downloaded. Hence the shorter term »Key« is used more often today. But in principle both, CD key or only key mean the same kind of information. On these platforms, however the license (and therefore also the key) is tied permanently to a platform account. After using it once, it simply cannot be used by anyone else any more/again. By authenticating to the platform account, you are able to install multiple games, whose license the account has.

Key Fraud/Theft

As such license keys are vital for the installation and claiming process, there is of course a certain sensitivity about them. Anyone who has a valid key can install the game, or in case of the above mentioned online platforms, claim the game for his own account. Especially the latter usually only works once. So once a key is used, it becomes useless for anyone else. Unused keys are sold legally on key reseller websites. But as some are actually already in use (which by principle you cannot derive only from the key itself, without actually using it), they are no longer valid for claiming by another person/account. This is also known as key theft or key fraud. It is a general draw back of key reseller sites. While mostly working properly, you might sometimes end up with a useless key anyways.

Also on games, that do not use the installation limitations imposed by online platforms (key is tied to an account), very often a black list mechanism is used, to prevent players from sharing keys. It is therefore important to keep keys to/for yourself. Otherwise your key might end up black-listed, or cannot be used/sold properly any more, as someone else already claimed it.

Black Listing Mechanisms and Effects

Concept

Mostly you simply cannot install a game without entering a valid CD key, or you cannot claim a game for an online platform without providing an unused, genuine key. In case of CD keys, it might suffice to provide *some* valid key, in order to proceed with the installation or using of the game.

There are also mixed implementations of this concept, where the installer checks for the general validity of a key and either the installer or the installed game also checks back with some sort of authoritative server, whether the key is black-listed or not. The process of claiming or entering the key is then also referred to as activation. Therefore the keys are sometimes also called activation keys.

Other games such as Unreal Tournament 2004 cut off certain aspects of the game, once the key is invalidated by black-listing. Very often that is the online multi player part. But it may as well be the entire game.

Concrete Example

Unreal Tournament 2004 uses a technique, that requires the game client (the game software itself) to provide the key without actually handing over the key itself and without the means to calculate the key reverse from the sent data. There are three things, that are necessary for the communication to be safe and yet easy to validate and not repeatable:

1. The authority side (in this case, the UT2004 master server) has access to a list of all CD keys that were actually handed out (=genuine), their MD5 hash sums and an information, whether a certain key is still valid (=blacklisting).
2. The game client needs to have a genuine key.
3. The server sends the client a random number, that the client has to use, in order to prove he actually possesses the key he wishes to use.

The concept works as follows: Epic keeps the list from point 1. *All* CD keys they ever gave out are listed there. From those keys the MD5 sum is calculated one time (when adding new keys issued) and stored in some sort of database, along with the key they belong to and an information flag, whether the key can still be used (=not blacklisted).

Whenever a client requests an information from the master servers of Epic, the first thing the master server does, is sending a so-called challenge. In case of UT2004 it is an integer number randomly chosen.

The client is now expected to send the following information for validation: The MD5 sum of the CD key the client wishes to use/is using. This allows the server to identify, which key is used, as it has a list of all hash sums and can simply filter the list for this particular hash sum and see one ore more keys that are leading to that hash sum.

With that information alone, anyone could do a replay attack, simply repeating the same hash sum, as someone else sent to the server. (The connection is not encrypted, so technically, anyone in-between can eves-drop on the communication and just repeat the same hash.)

In order to be able to validate whether the client actually does have the corresponding key, the client has to send a second information. In this case it is another MD5 sum, calculated from the CD key with

the just received challenge number added after it. As the random number changes every time but is expected to be used, this hash sum is always different (cannot be replayed, unless the same random challenge occurs).

The server can then do the same calculation. Use the challenge it just sent to the client, use the CD key(s) just derived from the other hash sum, and compare the results. This also allows for multiple keys having the same hash sum (MD5 can of course have collisions, however unlikely) without causing any trouble. With the second hash sum it is extremely unlikely to simply guess a proper CD key from the MD5 hash of the key and the challenge just required by the server. If no CD key is found, that matches the derived value, it cannot have been issued and therefore must be invalid.

As a last step, it may also be checked, whether the identified CD key is actually still allowed to be used. It may have been shared on the internet and being black-listed in turn.

This is a very simple and elegant way to not require anyone to send their real CD keys over the internet and still being able to check for the validity of an actually used key, incl. a check if it is no longer valid. So the vendor keeps control over the keys he issued.

This also shows, why CD key generators (»keygens«) may produce technically proper keys (accepted by the installer or game itself) by using the original algorithm used by the vendor, but which still aren't valid, as they don't appear on the lists of the vendor.

Games Database