On Repairing Broken Cryptographic Algorithms

Michael P. Abramson
William J. Layton

Abstract

We disclose a method for delaying the implementation of an attack on a cryptographic algorithm. This method may be extended to other physical security scenarios.

1 The Re-design of Algorithms

The cycle of designing, attacking and re-designing cryptographic algorithms is well known to cryptographers and is healthy for the development of cryptography. Attack techniques are refined, design principles are crystallized, and confidence is built in the security of tested algorithms. When algorithms are not subjected to such a process, smart users have less confidence [3].

When an algorithm designer faces the prospect of re-designing his algorithm because of an attack, he must carefully weigh all options. For example, consideration must be given for how devastating the attack is, how practical the attack is, and how much it will cost to repair the algorithm (especially in the case of a hardware implementation).

2 Patents

In this paper, we propose repairing a broken algorithm by patenting. For purposes of this discussion, we only consider U.S. patents, but other countries have similar laws. A patent is a contract between the inventor and the government which excludes others from manufacturing the invention in exchange for full disclosure of the invention. Enforcing a patent is generally left

to owner of the patent. The purpose of a patent is to encourage innovation and help technology, but they are always time limited so that technology can progress. U.S. utility patents are valid for 20 years from the date of application filing.

To obtain a patent, an invention must be useful, novel and not obvious. These three terms are subject of course to wide legal interpretation, but the basic idea is as follows.

- *Useful*: The invention must have a useful purpose and must work properly (a machine that does not work as it supposed to cannot be patented).
- *Novelty*: The invention is new; however, once the invention is made public, the inventor must file for a patent within one year in order to obtain a patent.
- *Not obvious*: The invention is new, and differences between it and prior inventions would not be considered obvious to someone with ordinary skill in the area of technology of the invention (changing color or size of an existing invention are normally not patentable).

For more information on patentability, consult the US Patent and Trademark Office web site [6].

3 Buying Time With Patents

Now let us consider several scenarios where cryptographic patents may come in useful.

Patenting Your Algorithm. The merits of patenting cryptographic algorithms have been widely debated. Companies want profits, and users want free security. Here's a way you can have both. Patent your algorithm, and license it for free to everyone, as long as they agree not to cryptanalyze it. Never license it to anyone who wants to cryptanalyze it. Make all users sign a license agreement. Now if your algorithm ever gets attacked, you can sue the attacker for violating the licensing agreement and probably for patent infringement since any attack will likely include implementing your algorithm.

Applying a New Attack Technique to Your Own Algorithm. In this scenario, a new attack A has emerged in the literature that may put

your already designed and fielded, publicly available algorithm X at risk. As a designer, you don't know yet whether your algorithm is at risk because the attack is too new. So like a good cryptographer, you immediately start to work, attacking your own algorithm with A to see if your algorithm is vulnerable.

If it is not vulnerable, then you breathe a sigh of relief, but if you find that your algorithm is vulnerable, you have a new invention, for which you apply for a patent. The title will be something like: *Recovering Lost Keys for Algorithm X Using Method A*. In the patent document, you disclose the attack A as a method to recover a lost cryptographic key for your algorithm X. This satisfies the usefulness requirement because it can be used to recover a key if the user forgot it (for patentability, it doesn't matter if there is a whole body of literature in key management or how long it takes to recover the key). Your method is certainly new because the new attack technique A has never been applied to your algorithm X. Furthermore, it will satisfy the non-obvious criterion because you will of course publish your results in a reputable publication, which a person with ordinary skill could never do. The advantage of patenting your attack in this way is that if someone decides to attack you, you can sue them for patent infringement, and potentially recoup all the money you would have lost if someone had disclosed the attack on your algorithm and you left it vulnerable. By patenting the attack on your algorithm, you have just postponed the necessity of re-designing your algorithm for 20 years.

Applying a New Attack Technique to Someone Else's Algorithm. If you are the first to apply a new attack on an existing algorithm that someone else designed, you could patent the attack and any (even all) countermeasures, and then refuse to grant the algorithm designer a license so that their algorithm remains vulnerable for twenty years, or they spend lots of money in court.

4 Extensions of Patenting

We can easily envision more general settings in which patenting might be advantageous. For example, new hacking techniques against your own computer network might be patented to discourage hackers from attacking your network. Of course, this would be obvious unless your network had some unique qualities that made it different from other networks. More cynically,

if you find a security problem in a major operating system, you might be able to patent the vulnerability and any countermeasures and then refuse to license it to the owner of the operating system.

Also, if a new type of bomb were invented, patenting it, rather than keeping it secret, would allow the owner of the patent to sue the users of the bomb for patent infringement if they ever used the bomb for terrorism (though the owner would probably never collect). One could also extend this to innovative methods of committing crimes or acts of terrorism [1, 2, 7].

5 Patenting Patenting an Attack

The title is not a typo. We are now telling the reader that we reserve the right to patent the method described above of repairing broken algorithms by patenting. Using a patent document as part of a patent is not new [4, 5].

References

[1] P. Belluck. *Crew Grabs Man; Explosives Feared*. New York Times, 23 December 2001.

[2] A. Cowell, D. Filkins. *British Authorities Say Plot to Blow up Airliners was Foiled*. New York Times, 10 August 2006.

[3] M. Curtin. *Snake Oil Warning Signs: Encryption Software to Avoid*. 10 April 1998. `www.interhack.net/people/cmcurtin/snake-oil-faq.html`

[4] R. Grace. *Method and Instrument for Proposing Marriage to an Individual*. United States Patent Application Publication 2007/0078663.

[5] G. Plow, F. Pourmirzaie. *System and Method of Extracting Value from a Portfolio of Assets*. United States Patent Application Publication 2007/0244837

[6] United States Patent and Trademark Office web site: `www.uspto.gov`

[7] G. Schmemann. *U.S. Attacked: Hijacked Jets Destroy Twin Towers and Hit Pentagon*. New York Times, 12 September 2001.