Lesson in a Haystack:
Idealists Take On the Theocracy

by M. E. Kabay, PhD, CISSP-ISSMP
Associate Professor of Information Assurance
School of Business & Management
Norwich University, Northfield VT

In mid-2009, Twitter [http://twitter.com/iran09] became a prime medium for communications within and out of Iran during the protests over the corrupted elections [http://twitter.com/iran09] in that fundamentalist theocracy [http://www.newsweek.com/2009/06/19/theocracy-and-its-discontents.html]. At about that time, San Francisco programmer Austin Heap [http://www.linkedin.com/in/austinheap] expressed his anger with repression by publishing lists of proxy servers that could be used to evade Iranian government censorship; the proxies promptly began disappearing and he received warnings from Iranians that the government was attacking the proxies. Matthew B. Stannard, San Francisco Chronicle Staff Writer, wrote [http://articles.sfgate.com/2009-06-17/news/17211048_1_proxy-server-twitter-iranian-government] that Heap’s next effort was “… creating a password-protected list of proxy servers and giving only a handful of people access to each, reducing the possibility of a widespread attack. On his blog, he published simple instructions for configuring proxy servers.” Within a few days, “his site came under a denial-of-service attack - a flood of phantom file requests from the United Kingdom designed to bring his system to its knees. Tuesday morning he received his first e-mailed threats.”

Heap and his online colleague Daniel Colascione from Buffalo developed Haystack [http://www.haystacknetwork.com/], which they described in their Frequently Asked Questions (FAQ) [http://www.haystacknetwork.com/faq/] page as “a computer program that allows full, uncensored access to the [I]nternet even in areas with heavy [I]nternet filtering such as Iran. We use a novel approach to obfuscating traffic that is exceptionally difficult to detect, much less block, but which at the same time allows users to security use normal [W]eb browsers and network applications.”

The project participants responded to the question, “Is Haystack secure” by writing, “Yes. We go to great lengths to ensure that any traffic between our servers and our users looks like perfectly normal, innocuous, and unencrypted [W]eb traffic. It would be exceptionally difficult to detect and block automatically.” They added, “However, even if our methods were compromised, our users’ communications would be secure. We use state-of-the-art elliptic curve cryptography to ensure that these communications cannot be read. This cryptography is strong enough that the NSA trusts it to secure top-secret data, and we consider our users’ privacy to be just as important. Cryptographers refer to this property as perfect forward secrecy.”

Unfortunately, they responded to the question, “In keeping the source code a secret, aren’t you just relying on ‘security through obscurity’? Won’t authorities eventually discover how your software works anyway?” with an insistence that “Everything that one of our users sends and receives is enciphered. It would take centuries for all the world’s computers to decipher one of our users’ browsing sessions even with full access to the Haystack source code.”

In the next article in this two-part series, we’ll watch as the Haystack collapses.

***
M. E. Kabay, PhD, CISSP-ISSMP, specializes in security and operations management consulting services and teaching. He is Chief Technical Officer of Adaptive Cyber Security Instruments, Inc. and Associate Professor of Information Assurance in the School of Business and Management at Norwich University. Visit his Website for white papers and course materials.

Copyright © 2010 M. E. Kabay. All rights reserved.

Permission is hereby granted to *Network World* to distribute this article at will, to post it without limit on any Web site, and to republish it in any way they see fit.