Security in an Open Electronic Society

A recurring argument in the computer security world is that clamping down on the dissemination of information about vulnerabilities, and tools that exploit them, will mitigate everyone's risk. Last week, one proponent of this argument, Scott Culp, manager of Microsoft's security response center, coined the term "information anarchy" to describe the current situation, comparing it with yelling "fire" in a crowded movie house.

It appears Culp is more comfortable with an "information dictatorship" or "information oligarchy" model, and has entirely missed the fact that the movie house ***is*** on fire.

A successful attacker requires three things: the opportunity to launch an attack, the capacity to successfully execute the attack, and the motivation to attack. An opportunity to launch an attack requires a vulnerable system and an access path to the system. The capability to successfully execute the attack requires knowledge of the vulnerability and the tools to exploit it.

Proponents of the information dictatorship argument are targeting the second requirement of a successful attacker: his capability to launch an attack. This approach to the problem of computer security is flawed, and can only fail.

First, we cannot stop some small number of malicious users from gaining knowledge of vulnerabilities, or access to the tools that exploit them. Vulnerability information and exploits have legitimate uses with the computer security field. They are part of research, are required in penetration testing, and used by system administrator to test their systems, mitigate the risks by gaining an in-depth understanding of the problem, and to verify that vendor fixes work as advertised.

We live in an open society. It is impossible to distinguish a potential attacker from a legitimate user of the information. Thus, it is impossible to compartmentalize the information and still make it available to everyone who legitimately needs it.

Playing Make Believe

The commonly proposed solution to this problem is to artificially restrict the information to a subset of legitimate users, marginalizing the rest. But this solution is no panacea either. History has shown that such groups will have voluntary, or involuntary, leaks proportional to the size of the group.

But lets put aside these problems. Lets make believe we can keep secrets. Lets pretend for a moment that the dubious claim that administrators don't need to know how a vulnerability works in order to protect against it is actually true. Have we solved the problem?

No.

This solution also suffers from the problem that it assumes only people within the privileged group are capable of finding vulnerabilities and writing exploits, or that anyone else that is capable of these tasks will only communicate with the group. This is a highly unlikely situation. A large number of people have this capability, and not all of them will agree with this policy, or have the same ethics.

Moreover, information on finding vulnerabilities and writing exploits is readily available and is part of the open exchange in computer security research. This information will not suddenly disappear, nor will it be banned.

Second, we cannot stop the small number of malicious users with knowledge of vulnerabilities or exploit tools from distributing them to other malicious users.

Vulnerability information and exploit tools are normally embodied as electronic documents. This allows the creation of copies at an insignificant cost. Malicious users are beneficiaries of the revolution in communications like the rest of us. They have efficient communication channels (IRC, web sites, mailing lists, instant messaging, etc). They also have a social structure based on trade, where vulnerability information and exploits are treated as a commodity. Thus, a single individual is capable of arming a large number of malicious users in a matter of seconds.

Security Scapegoating

One proposed solution to this conundrum is to outlaw disclosure of vulnerability information and transfer of exploits to anyone but legitimate users of them. Once again -- putting aside the problem of determining who is a legitimate user and who is not -- we still have the problems of freedom of speech, globalization, and anonymity.

The solution would infringe on the freedom of speech afforded in many places, although this is not an insurmountable problem. There are limits to what one can say, especially if the specter of "national security" is invoked. Such limits can also be legislated by industries, with enough lobbying, as Professor Felten and others learned when they were threatened by the recording industry under the DMCA.

For such a solution to have any teeth it would have to be implemented on a worldwide basis. That may not be such a far-fetched idea, given the Council of Europe's computer crime treaty, but it's not easily accomplished in real life. Laws are pretty useless unless they are actively enforced. The Internet and other technologies can provide a high degree of anonymity for those that know how, and that knowledge is highly valued by malicious users. If the unsuccessful searches for the authors of many Internet worms and viruses are any indication, would-be law-breakers would have little to worry about by exchanging vulnerability information and exploits with other malicious users.

From these facts it should be obvious to most observers that attempting to degrade an attacker's capability to execute the attack is a losing battle. Instead we should focus our limited resources on denying would-be attackers the *opportunity* to launch an attack, and on neutralizing their motivations to attack.

Of course, ethical folks should try to disclose vulnerabilities and tools in a responsible way. But just what is responsible in an open society that values research and the open exchange of ideas is open to interpretation and is different for each person.

While we don't yet have the technology to develop bug free software or hardware, there are plenty of examples that demonstrate that a commitment to develop a secure product throughout the design, implementation, and deployment phases can dramatically reduce a malicious user's opportunity to attack. It's high time for vendors of vulnerable products to clean up their act and stop looking for scapegoats for their lousy products.