Full Disclosure of Vulnerabilities – pros/cons and fake arguments

Arne Vidstrom April 8, 2002 Share

Should the complete details of security vulnerabilities be made public or not? Not only do we need to understand the true pros and cons, but we also need to understand the "fake arguments" – the arguments people bring forth to serve some other purpose than making the "truely right" decision. This paper will try to point out all these things, to aid in building a more complete picture of the full disclosure concept.

What should be the restrictions for full disclosure?

Some restrictions should probably be:

Ã‚Â» The vendor should be given a reasonable chance to provide a patch or new version before the vulnerabiliry details are made public.

In some cases the system administrator may be able to fix the problem without a patch – in these cases there would be no necessary need to wait for a vendor patch. A thing to remember though, is that there may be compatibility problems preventing some administrators from applying "quick fixes". The vendor usually takes these things into account when creating the patch.

The vendors need to provide sufficient information to the public so finders of vulnerabilities know how to contact the vendors. Of course they also have to really look at the vulnerability reports. I personally have experienced vendors who reply that they will not consider my findings because I am not registered as a customer...

Ã‚Â» When releasing the vulnerability details they should be released completely. The attackers usually have a lot of spare time to figure out the

missing parts, but the busy administrators usually don't.

Ã‚Â» The vulnerability details should at least be published at places where they reach the largest possible group of security people, for example NTBugtraq for Windows NT / 2000 related bugs. Publishing the information only on not so well known places, increases the risk that attackers will use it before anybody has the chance to fix the problems.

Which are the pros?

Ã‚Â» If the vendors know that complete vulnerability details have been, or soon will be, made public they hurry up creating patches.

There is however a risk that the vendor will be stressed to release a patch before it is really thought through and tested. The patch may then not fix the problem completely, or cause compatibility problems.

Ã‚Â» If an administrator knows that there are complete vulnerability details made public, this increases the chances that he/she will take the problem seriously and really apply the provided patches.

There are many reasons for an administrator not to apply all available patches. They include worries that the patches will introduce new errors into the system, a high work load, plain lazyness, and that patches for example the OS are not fully supported by application program vendors. Knowledge about the fact that vulnerability details are in circulation out there also gives the administrator an argument against management/vendors for more resources in security issues.

Ã‚Â» Those who create security scanners need as detailed descriptions of new vulnerabilities as necessary.

If an "outsider" keeps them secret, there is an increased and unnecessary work load (and delay) for these vendors. If one of them has and keeps the details secret, there is an increased work load (and delay) for the others.

Of course this could be considered as a benefit due to competition (bla bla...). But do you really want the vendors competing while you wait for a working scanner to test your systems, when you know that there may be attackers out there who exploit those vulnerabilities?

Ã‚Â» Those who do penetration testing need as detailed descriptions of new vulnerabilities as necessary.

How could we keep them secret when so many people at so many different places and organizations have them?

Ã‚Â» The vulnerability details may already be in circulation in the "computer underground". Full disclosure will even the odds in the battle between attackers and defenders in this case. There is a slight chance that somebody knowing the details will publish them for personal fame, but there is not a chance that any of the bad guys will give them to the vendors just to be nice.

Ã‚Â» If the vulnerability details are published, all the developers in the world can learn from them and try not to make the same mistakes again.

Ã‚Â» An attacker usually has a lot of spare time to find out the details of a new vulnerability, but an over-worked administrator usually doesn't. Full disclosure will in this case be more of benefit to the administrator than to the attacker.

Ã‚Â» An administrator might use vulnerability details to find similar vulnerabilities in other systems. If the same administrator only has a patch that "does something to increase security", he/she will not have a chance to notice that a similar vulnerability exists in another system.

Which are the cons?

Ã‚Â» If the vulnerability details had not been published, there might not have been a single attack performed using the specific vulnerability.

The problem is that you can never be sure of that until there are no more affected systems in use anywhere. There is also a great risk that the really dangerous attackers are never noticed, because they do not deface web sites or other obvious things.

Ã‚Â» A lot of script-kiddies without any detailed knowledge can now exploit the vulnerabilities. If the details had been kept secret, they would not had the resources to do it.

The script-kiddies are really only a decoy in the security game. Since they by definition don't have deep knowledge about security or computer systems they are limited to simple attacks. Your most sensitive information and systems should not be "hackable" through just a single, simple, exploit. Script kiddies usually do nothing more than deface web sites and similar, even when equipped with complete exploit details. The attackers who are capable of more dangerous attacks, will most likely be able to create their own exploits without the complete vulnerability details. Having the external web site defaced will of course be a bad thing, but having the most sensitive secrets stolen and given to a competitor may be a lot worse, right?

Ã‚Â» The vendors may be stressed to provide patches that are not really thought through and sufficiently tested. That may even be worse than providing no patches at all in some cases.

Ã‚Â» When vulnerability details are first published, there will be a flood of attacks against affected systems.

What is behind the "fake arguments"

There are inevitably a lot of arguments brought forth which seems to be focused on "right and wrong", but which in fact have other underlying reasons. In this section we will look at a few of those reasons.

Ã‚Â» Money – the vendors simply think they will make more money from keeping the vulnerabilities secret.

A poll among the ntsecurity.nu visitors was the following:

Question: Do you think that software vendors deliberately neglect security to increase short-term profit?

yes – 87%

no – 7%

not sure – 6%

If we don't trust the vendors, we need some kind of balancing force – for example full disclosure.

Ã‚Â» Personal fame – "disclosing complete vulnerability information with working exploit code will make me more famous".

Of course this is one of the driving forces behind people making vulnerability details public, but it would be stupid to think it is the only reason, and that there are no "good" reasons. Neither do all people necessarily have this as one of their reasons.

Ã‚Â» Control – "if I keep the information secret I will be in control, me and my elite security expert friends will not allow anybody else to enter our closed elite group".

Ã‚Â» Once again, money – "if all vulnerability information is kept secret, our company doesn't have to spend any money on security".

Right, and when you are hacked anyway you'll just pretent it never happened...

Ã‚Â» Lack of knowledge – “if I don’t understand the explicit vulnerability details even when I have them in front of me, I sure as hell don’t want anybody else to have them!”.

Other aspects

Ã‚Â» Who decides who should be allowed to know the vulnerability details and have the exploit code? The big companies, the government, the researchers? Will those people serve the best interest of the society as a whole, or of themselves?

Ã‚Â» Shouldn’t you be allowed to have all the available information concerning the security in your systems?

Ã‚Â» Shouldn’t the vendors have the final responsibility? After all, they design the systems to make money.

Ã‚Â» The vulnerabilities are already there! The people who find them and publish the information don’t create the vulnerabilities!

Ã‚Â» Companies with sensitive information must be prepared to spend money on security. If they can’t afford it, their business isn’t profitable enough in the first place.

Conclusions

There is a time for full disclosure, and a time for covering things up, it all depends on which serves you best. “right and wrong” can be found on both sides, and in the world of computer security it is often not the thing people really focus on.