OPEN SOURCE SECURITY

February 2008

Disclaimer: Whilst the Government endeavours to ensure the accuracy of the information in this paper, no express or implied warranty is given by the Government as to the accuracy of the information. The Government of HKSAR accepts no liability for any error or omission arising from or related to the use of the information.

TABLE OF CONTENTS

SUMMARY

It is often claimed that open source software is intrinsically more secure than closed source or proprietary software. Others argue that it is not, and it is expected this debate will continue for some time to come. The availability of source code provides both attackers and defenders opportunities to study code in detail and identify software vulnerabilities.

On the other hand, closed source software forces users to accept only the level of security diligence that the vendor chooses to provide. This paper discusses ways in which we can take advantage of the nature of open source software with regard to IT security. We also outline a number of best practices in open source software security that are recommended by the open source community, along with important points on using open source products safely within the organisation.

I. INTRODUCTION

WHAT IS OPEN SOURCE SOFTWARE?

Open source software usually refers to software whose source code is “open” and available to anyone to study, use and adapt. According to the Open Source Initiative[1], the terms for the distribution of open source software must also comply with 10 criteria specified in the Open Source Definition[2]. The top 3 items out of the 10 criteria include:

1. software should be freely redistributable,
2. software must allow for distribution as source code as well as in a compiled form,
3. licences must allow modifications and for derivatives generated from the source code.

Lists of software licences that comply with the Open Source Definition are available at Opensource.org[3]. Examples include the Apache Software License, the GNU General Public License (GPL), the IBM Public License, and the Microsoft Public License (Ms-PL).

[1] http://opensource.org/

[2] http://opensource.org/docs/osd

[3] http://opensource.org/licenses

The term freeware refers to software that can be used with no cost. Open source software is essentially freeware, but freeware software does not always make the source code available publicly.

USAGE TRENDS WITH OPEN SOURCE SOFTWARE

Open source software has been gaining in acceptance more recently, even in enterprise environments. In 2006, Unisys predicted that open source software would continue to gain acceptance from enterprise customers as a vehicle for deploying enterprise applications that are able to drive business growth and innovation at a lower cost per transaction[4].

In Europe, open source is considered a means of improving the competitiveness of the ICT sector[5]. In fact, as early as 2005, it was reported that nearly half of all European local government bodies were using open source software in some form[6].

OPEN SOURCE VS COMMERCIAL SOFTWARE

One distinct difference between open source and commercial software is the availability of source code for review. Because the source code for open source software is publicly available, it can be used basically for free. Many organisations, in particular small- and

[4] http://www.unisys.com/about__unisys/news_a_events/11288732.htm

[5] http://blogs.the451group.com/opensource/2007/11/22/europes-open-source-opportunity/

[6] http://www.theregister.co.uk/2005/10/21/opensource_government/

medium-sized enterprises, have chosen or are considering choosing open source software for economic reasons.

The free and open availability of source code is also considered to be an aid to software security because community-based peer review of source code can more rapidly help identify bugs or vulnerabilities in software. However, not everyone agrees with this argument.

Commercial software is mostly “closed source”. That is, the source code is not publicly available. Because the source code is not available, there is a barrier against access to the code that attackers have to cross, resulting in less likelihood of vulnerabilities in the source code being exploited even though vulnerabilities do exist. Again, not all people agree on this. After all, an unreported or unidentified bug does not mean that a flaw will go away.

There is not yet any universal agreement on whether open source security is better than closed source security, or vice versa. Arguments on both sides are compelling[7] and it is expected that this debate will continue for some years.

[7] http://www.acmqueue.com/modules.php?name=Content&pa=showpage&pid=453

II. USING OPEN SOURCE SOFTWARE AS A SECURITY TOOL

A variety of security tools have been developed by the open source community. The most popular use of open source security tools in the industry can be categorised as follows:

1. Firewalls, such as iptables.
2. Intrusion Detection Systems, such as Snort.
3. Network Monitoring Tools, such as Multi Router Traffic Grapher (MRTG).
4. Security Assessment Tools, such as Nikto for web server scanners.

As there is no official support for these open source tools, the use of such software carries inherent risks. Special care should be exercised, and management approval should be obtained before they are deployed in the organisation.

III. SOFTWARE SECURITY FOR OPEN SOURCE SYSTEMS

As discussed earlier, one characteristic of open source software is the public availability of source code, including potential criminals and attackers. Attackers are able to study source code and exploit vulnerabilities that may be due to programming flaws much more quickly. In addition, open source applications are usually developed jointly by volunteer contributions from groups and communities over the Internet. Attackers might also be able to contribute parts of the code to the software this way. Code level security usually depends on reviews conducted by those entrusted with maintaining the project or other contributors. However, it should be noted that closed source software could also suffer from similar problems if source code is leaked out to the public, such as the introduction of backdoors by disgruntled staff.

Efforts have been made by the open source community to improve software security and quality so as to mitigate vulnerabilities in applications and systems, including open source software. In general, open source software security is best achieved by following these best practices[8]:

1. Maintain an inventory of all software being used, including open source software. The software inventory should also document the version, the hash value (such as MD5 or SHA-1) for verification of the integrity of the source code, as well as the website where the software was originally downloaded.
2. Check the availability of security updates and bug fixes for open source software regularly so that patch management processes can be followed regularly to minimise any loopholes in the selected open source software.

[8] http://searchsmb.techtarget.com/tip/0,289483,sid44_gci1271530,00.html

3. Change all default security settings in open source software as soon as it is installed. Configure the product in the most secure way possible by disabling unwanted services.

4. Test and scan the source code with code analysers or auditing tools, such as BOON (Buffer Overrun detection)[9], FlawFinder[10], RATS (Rough Auditing Tool for Security)[11], and so on. Developers may also want to run compiler-integrated tools, such as ProPolice (or Stack-Smashing Protector, or SSP) from IBM[12], which automatically insert protection codes into source code that protect compiled programs[13].

5. Ensure that the open source application fully complies with existing network architecture if the application requires the opening of any firewall ports. This avoids any violation of the organisation's firewall and security policy when a new application is introduced.

[9] http://www.cs.berkeley.edu/~daw/boon/

[10] http://www.dwheeler.com/flawfinder/

[11] http://www.fortifysoftware.com/security-resources/rats.jsp

[12] http://www.trl.ibm.com/projects/security/ssp/

[13] Cowan, C., "Software Security for Open-Source Systems", Security & Privacy Magazine, IEEE, Volume 1, Issue 1, Jan.-Feb. 2003 pp.38-45.

IV. HOW TO USE OPEN SOURCE PRODUCTS SAFELY IN THE ORGANISATION

To use open source products safely, organisations must consider the following:

1. Set up a well-documented security policy and ensure the policy is strictly adhered to. This policy should be revised, as business needs change.

2. Download open source products only from trusted sites, such as the official website of the software developer(s), to avoid potential risks from pre-inserted malicious code.

3. Download source code rather than a compiled package. In this way, source code can be verified against the MD5 / SHA-1 checksums provided, analysed for security vulnerabilities and compiled for the organisation’s specific needs.

4. Study the product’s documentation carefully for any explanation of the secure configuration parameters.

5. Check whether there is a reporting procedure should a vulnerability be discovered in the product, and ensure all security issues around the product are well maintained and addressed.

6. Check regularly on common security vulnerability databases, such as CVE (Common Vulnerabilities and Exposures)[14], for published information on any security vulnerabilities pertaining to the open source product(s) being used.

7. Adopt a “Defence-in-Depth” strategy so that various threats at various levels right from the open source product to the network can be fully addressed.

8. Provide appropriate training to in-house staff for the support and maintenance of open source products. Put together proper documentation for all the practices and

[14] http://www.cve.mitre.org/about

configurations required in order to avoid problems that might arise due to job rotations or employment termination.

V. CONCLUSION

The adoption of open source software within an organisation is not as simple as just downloading and running a free program from a website. There are a number of security concerns that should be studied, weighed up and determined before an organisation takes the plunge into the open source world. In addition, both individuals and organisations need to keep in mind recommended best practices put out by the open source community. Organisations considering using open source solutions in the enterprise should be aware of all the points outlined in this paper.