Zero-Day and Less-Than-Zero-Day Vulnerabilities and Exploits in Networked Infrastructures[1]

Hartmut Pohl

All computers are at risk from security vulnerabilities that are generally unknown – to the user and even to the manufacturer. When a new vulnerability is identified, it is reported to the relevant manufacturer of the hardware and/or software who works to produce a fix in a reasonable time. Today, however, vulnerabilities are often dealt with in a different way. The person who discovers a vulnerability often sells or even auctions this knowledge to the highest bidder, and perhaps other parties later. This may or may not include the manufacturer – which poses a serious threat to all the users who are unaware of the (undisclosed) vulnerabilities.

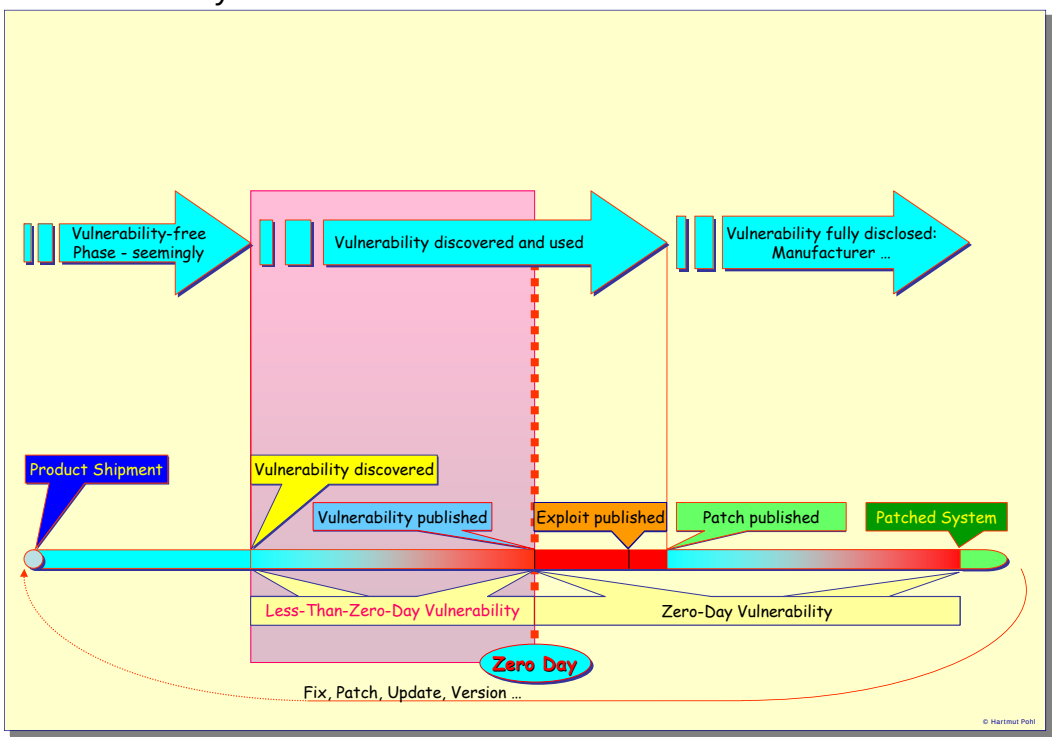
Less-Than-Zero-Day vulnerabilities and exploits may be bought by national agencies in order to fight organised crime by penetrating the computers of suspects such as criminals and terrorists. But these exploits may also be sold to organised crime, (foreign) intelligence agencies and companies. These may use the exploits for economic espionage and computer sabotage against other companies; a further real risk is the manipulation of Internet Top-Level-Domains and Routers. Several attack cases such as 'Titan Rain' have been published.

Lifecycle of vulnerabilities
The lifecycle of vulnerabilities can be divided into three phases – see figure below.

- During the first phase there are no vulnerabilities discovered – the software seems to be bug-free.
- During the second phase at least one vulnerability has been identified and possibly exploited.
- Only then in the third phase is the vulnerability disclosed and the manufacturer must develop a patch.
- **Vulnerabilities** – Today it is impossible to develop software to be completely bug-free. The vulnerabilities in IT systems (hardware, software such as operating systems etc., application software and even security programmes including firewalls and anti-virus programmes) may be discovered accidentally, but mostly they will be identified systematically with tools and proved with appropriate attack software (exploits) – as a proof of concept – and communicated to the manufacturer. Depending on the relevance of the vulnerability, the manufacturer develops a patch and publishes it – usually along with the vulnerability.
- **Zero-Day Vulnerabilities** and Exploits – The day of publishing the vulnerability is called Zero-Day: attackers benefit from the timeframe between the publishing of the vulnerability and patch-installation to

[1] Published in: ENISA Quarterly Review Vol. 4, No. 2, Apr-Jun, 7 - 9, 2008

penetrate systems by so-called Zero-Day Exploits. 25% of exploits are developed within 24 hours after publication of a vulnerability and 31% inside six days.



Caption: Lifecycle of vulnerabilities

- **Less-Than-Zero-Day Vulnerabilities and Exploits** – The time between the diagnosis of a vulnerability and its public disclosure is called Less-Than-Zero-Day. With the disclosure of a patch, the lifecycle starts again similarly to the original product shipment. In theory, Less-Than-Zero-Day Vulnerabilities can be attacked successfully by the corresponding Less-Than-Zero-Day Exploits: there are no specific security measures or tools for unpublished vulnerabilities; the victim especially is not in a position to recognise the attack directly.

  Security programmes such as anti-virus software, intrusion detection or intrusion protection systems are not really useful because they only audit attacks with known properties such as bit-string (signature) or behaviour (heuristics).

  In the first half of 2007, 90% of the 3.273 published vulnerabilities could be exploited remotely (over the Internet); more than 50% of these vulnerabilities provided access control rights for administrators; equivalent numbers probably apply to Less-Than-Zero-Day Vulnerabilities.

- **Vulnerability Market –** Some vulnerabilities are published via mailing-lists or other lists, resulting in increasingly fewer vulnerabilities being communicated directly to the manufacturer. Auctions for vulnerabilities exist on the Internet. Thus there is an open market. Some exploits are sold covertly to agencies, security agencies or individuals for economic

espionage or for organised crime. Some exploits are actually researched on behalf of such criminals or for botnet-providers; this underground market in particular is developing very quickly.

Attack Approach

- **Exploits** – Exploits are software-specific. Therefore the victim's software must be known by version, update, build etc. Exploits are also specific to a special vulnerability; there may be several exploits for one vulnerability. There are exploits which take advantage of a flawed programme directly, by gaining access rights, and exploits which begin by implanting executable code.

- **Adjustment to the Target System** – If the system's configuration consists of several computers (a multiprocessor system with servers, sequentially configured firewalls etc.), the exploit used must be determined for each computer and each computer must be attacked successfully by different exploits. A business model is proposed for discovering security vulnerabilities in complex hardware/software configurations. However the attack could use an open port into the whole system.

- **Attack Target** – One of the purposes of an attack is to gain increased access rights, e.g., to acquire the access rights of the system administrator, to read and write data; to combat this, rootkits may be installed to provide a virtualisation layer between the hardware and the operating system or the operating system and the application software, to process data in an unrecognisable way which cannot be audited.

- **Lack of Protection** – In fact published exploits may be recognised by security software, .e.g., anti-virus tools, but this is not the case for the newest, still unpublished, vulnerabilities and exploits and only applies actually during the attack. Using, for example, buffer overflows, code injections, viruses, worms or Trojan horses, eventual traces, e.g., in audits, can be covered and the software restored, so the victim is not able to recognise he has been attacked, let alone verify it.

Vulnerability detection tools and tools developing exploits

These tools support vulnerability detection and the development of exploits, for example: fuzzers are tools generating random input data systematically in order to launch a brute force attack; bugs and vulnerabilities which have not yet been detected by code-audits may be discovered by the misbehaviour of the software.

The Metasploit Framework (www.metasploit.com/framework/) is a development platform for creating security tools and exploits. The framework contains information on vulnerabilities, presents a tool to develop and use exploits and contains a shellcode-archive. It is used by network security professionals to perform penetration tests, system administrators to verify patch installations, product vendors to perform regression testing, and security researchers world-wide.

Measures to avoid or detect attacks
There are ways in which organisations and individuals can limit the possibility of becoming a victim of the exploitation of unknown vulnerabilities, including some which are not yet widely known. To reach a reasonable and adequate security level for IT systems to satisfy normal protection requirements, access control, firewalls, anti-virus programmes, anti-spam/spyware and intrusion detection and intrusion response systems are essential.

Application software can also be tested in various ways, statically and dynamically, with or without the support of a tool, and the source code can be checked for security bugs.

- **Internet access policy and data transfer rate** – Organisations and individuals should implement a restrictive access control policy for Internet access. Systems should be shut down or disconnected from the network when not in use; systems containing highly sensitive information should never be connected to a network.

  Limiting factors for attacks are a restrictive Internet access policy enforcing the 'target' system or physical shutdown when the system is not in use (sleep-mode is not adequate); furthermore the transfer rate of the target system is a limiting factor to attacks. More secure are standalone systems which have been physically disconnected from all networks including LAN, intranets, extranets or the Internet.

- **Security quality of software used** – In principle, software, especially security-relevant software – and hardware too – may merit a higher security level if evaluated and certified according to ISO/IEC 15408 (Common Criteria); this applies especially for Evaluation Assurance Level EAL 4 and higher.

- **Patch-Management** – Known vulnerabilities and exploits should be eliminated by a bypass, fix, hotfix or patch. Automatic remote maintenance and automatic updates should be avoided.

- **Storage Media** – Where possible, sensitive information and bootstrap information should be stored on portable devices. These devices would only be connected to the system when this is required. Attacks can also be hindered by using portable media such as discs and sticks to boot and store sensitive user and system data, as this can minimise their time spent online, creating a sandbox-like situation: such an environment may not be manipulated.

- **Hardware** – Where the attack exploiting the vulnerability is intended to copy substantial volumes of user or system data from the system, then the speed of the system's components (processors, memory and peripheral devices) and the network transfer rate may inhibit the attack or make it more likely to be detected.

- **Checksums and Audits** – Thorough examination of the system usage information may identify system actions that were unusual or

unacceptable. If the attack exploiting the vulnerability seeks to amend (or destroy) system or user data, this may be detected. In addition, appropriate (end-) user procedures should exist to detect changes in user data and /or programmes performing user functions.

Political Proposals

Developers may sell Less-Than-Zero-Day exploits to agencies such as secret services to penetrate the computers of suspects, criminals and terrorists, but they may also sell them to companies for economic industrial espionage, or even to cybercriminals.

A strategy is required to address this situation. The following action is recommended:

- Make public all the vulnerabilities and exploits of which agencies are becoming aware.
- Establish a European framework for exchanging information about unpublished vulnerabilities and exploits in real-time. A speedy exchange would enable companies to safeguard themselves against computer espionage and individuals to maintain their privacy.
- Develop a systematic approach for researching vulnerabilities in software to ensure the most widely used software is made more secure.

Providing better information to users, particularly home-users, as to how to detect when they have become a victim of the exploitation of vulnerabilities would also be beneficial.

The author is grateful to Willie List and Kai Rannenberg for their comments during the preparation of this article.

Prof. Dr. Hartmut Pohl (Hartmut.Pohl@fh-brs.de) is a professor of Information Security in the Computer Science Department of the University of Applied Sciences Bonn-Rhein-Sieg and is the CEO of the spin-off softScheck .