Kaspersky Lab report: Evaluating the threat level of software vulnerabilities

Overview

Vulnerable programs are among the most commonplace ways to attack victims and steal personal data. Exploits, pieces of malicious code that utilize vulnerabilities in popular software to infect the system, are used in malware designed to steal consumers' personal data, but they are also the philosopher's stone of cybercrime wizardry in terms of targeted attacks or cyber warfare. All known cyber weapons, such as Stuxnet and Duqu, used exploits to sneak into heavily guarded IT infrastructures for the purposes of sabotage and cyber espionage.

The main goal of Kaspersky Lab's team of security experts and analysts is to identify and block all new cyber threats, including exploits. Apart from the traditional methods of detecting and blocking particular malware samples based on their signatures, new, smart techniques are used to block even previously unknown exploits or those that utilize newly discovered, or "zero-day", software vulnerabilities. Automatic Exploit Prevention is a prominent example of this innovative technology. It detects and blocks exploits based on their behavior, before they can harm our customers. To develop these kinds of technologies, we need to really understand what our customers need: which programs they use and how they deal with vulnerable software.

We compile this data using the cloud-based Kaspersky Security Network: in exchange for this invaluable information our customers benefit from this network by receiving the most up-to-date news on the latest threats in almost real-time mode. Before coming to Kaspersky Lab's servers, the information about local security incidents and usage data is cleaned from all personal information, maintaining strict anonymity.

This report is based on information about vulnerable programs found on the computers of our customers. The vulnerability scan is one of the standard features of Kaspersky Lab products like Kaspersky Internet Security 2013: it helps users to identify and upgrade critically vulnerable software. The purpose of this research is to understand how users react to vulnerable programs and analyze the potential dangers of vulnerable software.

Methodology

- The source of data: customers using Kaspersky Lab consumer security products who agreed to join Kaspersky Security Network
  - Data was collected only from Windows-based PCs
  - Total number of users: over 11 million
- Period of analysis: January to December 2012, on a weekly basis, 52 weeks in total
- General analysis of all vulnerabilities was performed, with the following criteria:
  - Year when the vulnerability was first discovered
  - Severity level
- The 37 most dangerous vulnerabilities were selected with the following criteria:
- More than 10% of users had software with this particular vulnerability during at least one week of 2012. From these we chose eight vulnerabilities that are actively exploited by cybercriminals and analyzed them more closely, seeing how their relative prevalence changed over the course of the year.

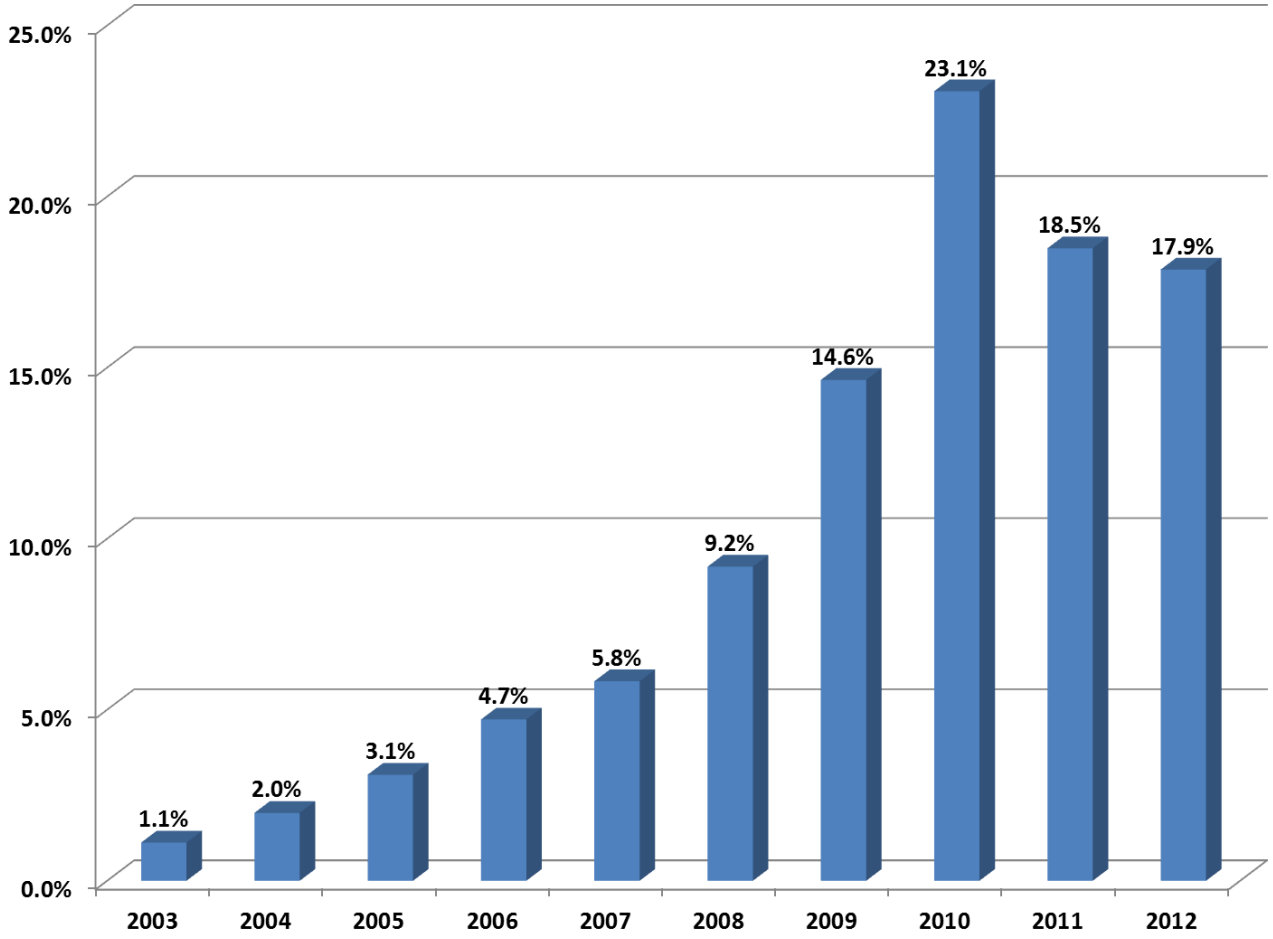
- Additionally we analyzed usage patterns for Oracle Java, using anonymous Kaspersky Security Network data on the versions users actually launched in September and October 2012.

Main findings

- Over 132 million vulnerable applications were recorded in total
  - An average of 12 vulnerabilities per user
- 806 unique vulnerabilities were found. Only 37 were found on at least 10% of computers during at least one week of the analysis period. These are the vulnerabilities likely to attract the cybercriminals' attention.
- These 37 vulnerabilities are found in 11 different families of software. The software with the greatest number of vulnerabilities includes Adobe Shockwave/Flash Player, Apple iTunes/QuickTime and Oracle Java.
- Further analysis of this list revealed only eight vulnerabilities that are routinely used by cybercriminals in widespread exploit packs.
- Oracle Java vulnerabilities have the highest impact: five out of those eight actively exploited vulnerabilities are found in Java Software. Another two belong to Adobe Flash and one more is found in Adobe Reader.
- The average threat level for all 37 top vulnerabilities is 3.7. That is calculated based on the severity level of each vulnerability, and falls between "Moderately Critical" and "Highly Critical".
- **The most alarming finding from this research is that users of the three most vulnerable programs (Java, Flash Player and Adobe Reader) are highly reluctant to update to newer, safer versions. A further look at the real use of Oracle Java showed just how bad this situation is: seven weeks after the release of a new version less than 30% of users had upgraded, despite being in real danger of having their data stolen. Major web browsers take only 5-7 days to achieve similar market share for newly released updates.**

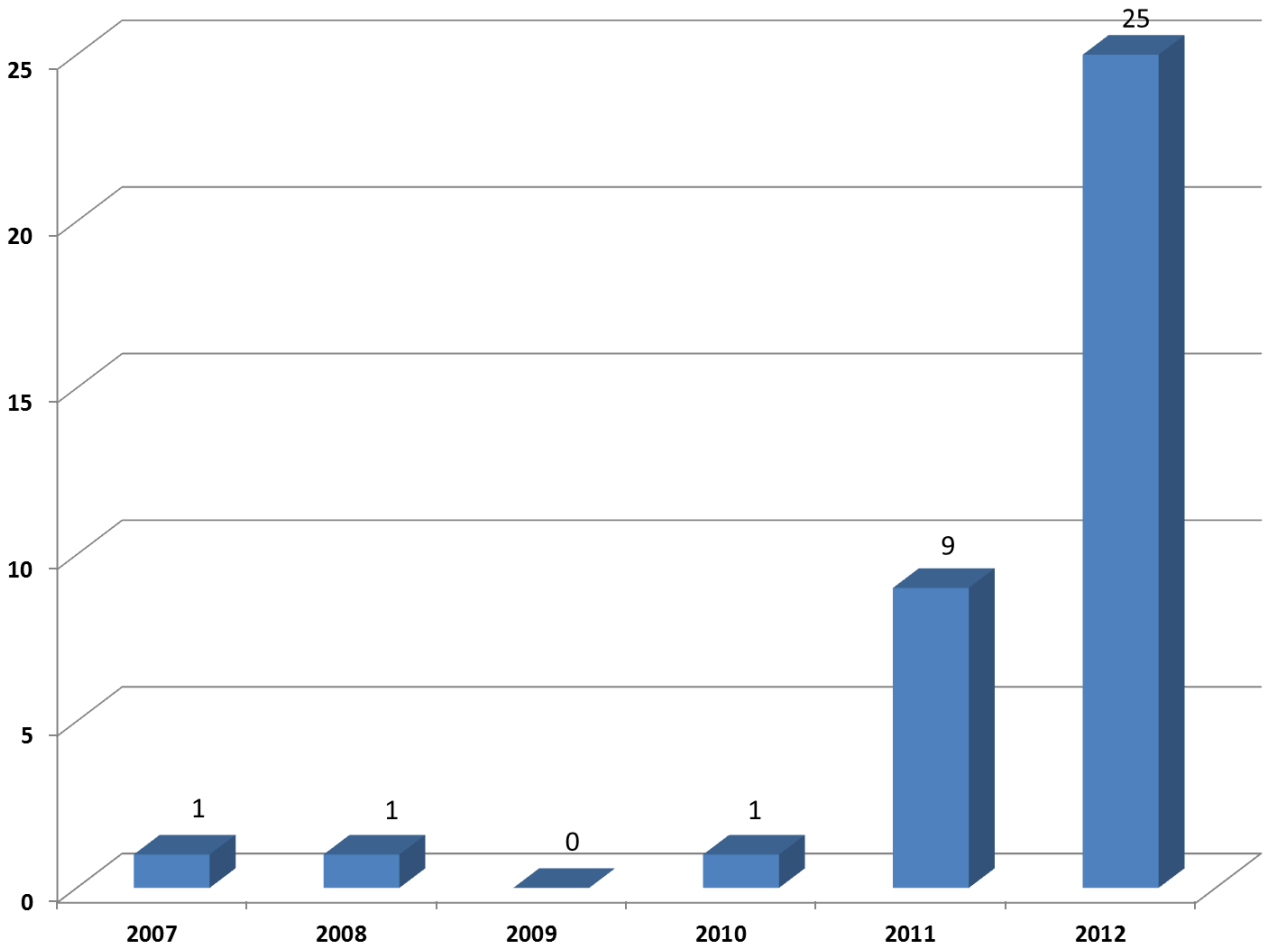
General figures

In the 52-week period we detected a total of 806 unique vulnerabilities on our customers' PCs. The oldest of them was first identified in February 2003; the most recent was in December 2012.



The share of vulnerabilities by year of discovery, all vulnerabilities

The best strategy to avoid potential security risks related to vulnerable software is to keep all your programs up to date (although this alone is not enough). The age of these vulnerabilities shows that users are failing to do this, except in those few cases where a vendor has been reluctant to issue an update. Sometimes, of course, everyone forgets about a rarely used program, or turns off irritating notifications. Analysis of the discovery dates for all vulnerabilities paints a grim picture: almost two-thirds (64%) of discovered software flaws are found in programs which are more or less obsolete (released in 2010 and earlier). But in order to get a clear picture, we need to take the "popularity" of certain vulnerable programs into account. To do this we counted only those vulnerabilities that were found on at least 10% of computers at some point during the year.



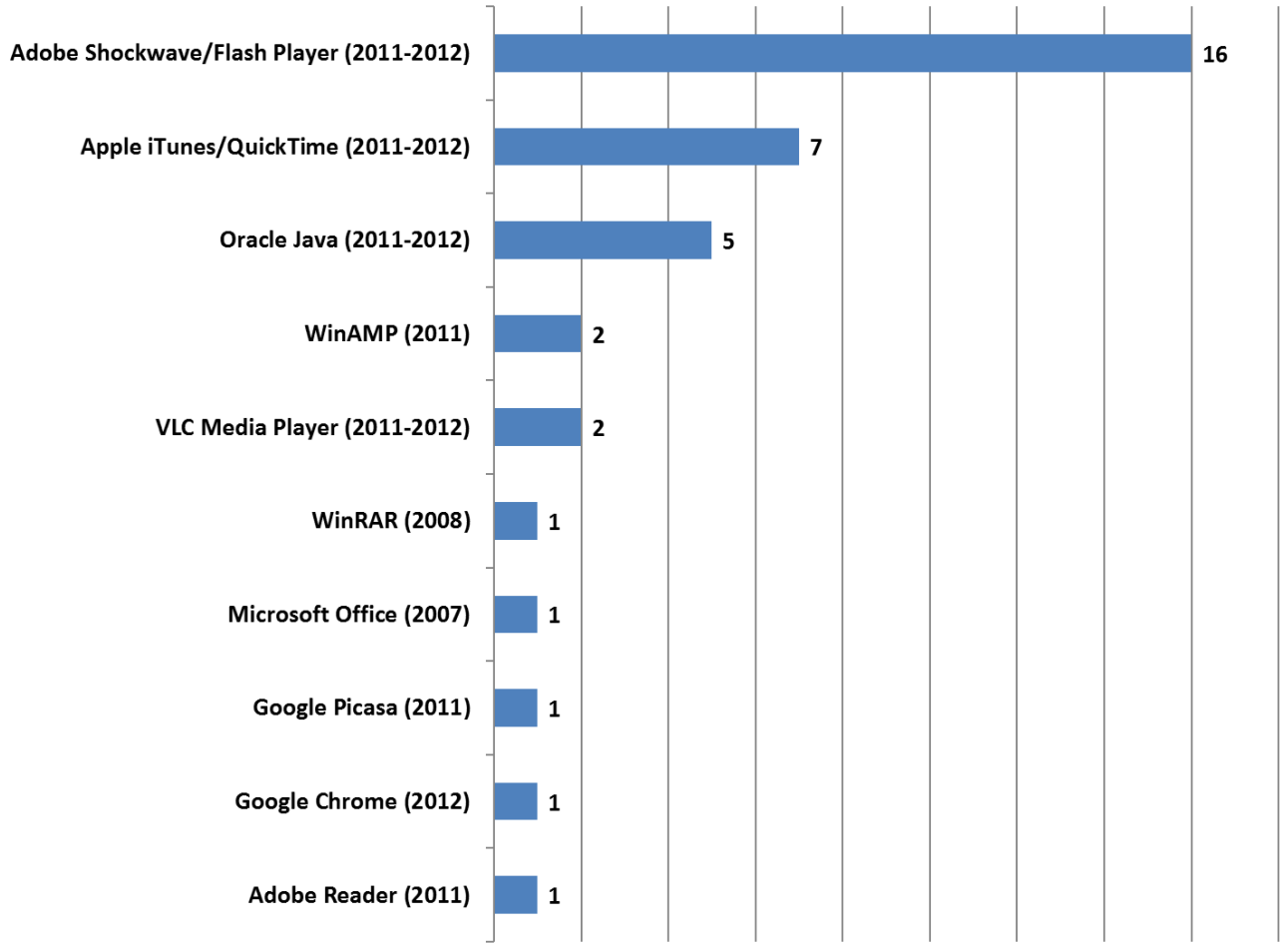
The top vulnerabilities by year of discovery

And here you can really see the difference. Only 37 vulnerabilities were sufficiently widespread to pass our artificial filter. The breakdown by age is also rather different: the overwhelming majority of popular vulnerabilities were discovered in 2011 and 2012, and only three vulnerable programs originate from 2010 or earlier (among the most notable being the vulnerability found in Microsoft Office 2007).

- **The 37 vulnerabilities from this chart in fact account for over 70% of all detections of vulnerable software during 2012.**

But one should keep in mind that hundreds of rare vulnerabilities could still potentially be used in targeted attacks on businesses.

Affected software



Differentiation of top vulnerabilities by software families. For each software family the number of top vulnerabilities is given, along with the time period when those vulnerabilities were discovered

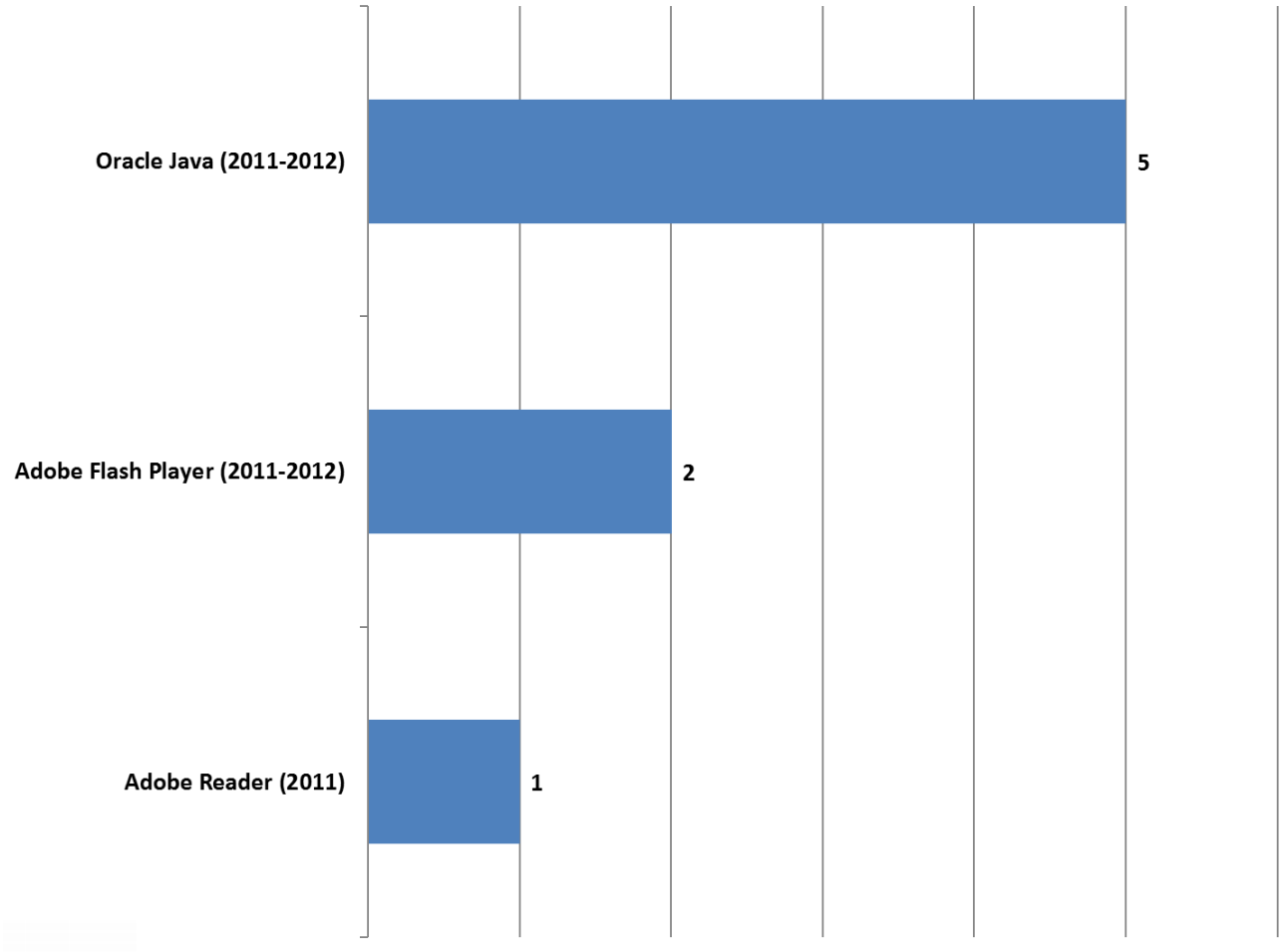
The top 37 vulnerabilities are found in 10 different product families. The most vulnerable products are Adobe Shockwave/Flash Player, Apple iTunes/QuickTime and Oracle Java. Between them, they account for 28 vulnerabilities among those found on 10% or more of users' PCs during 2012.

Threat level

One of the most important characteristics of a vulnerability is its severity. In Kaspersky Lab's vulnerability database the lowest severity is 1 (not critical) and the highest is 5 (extremely critical). Vulnerabilities with severity level 5 are considered to be the most dangerous, as they theoretically can be easily exploited and are most likely to lead to the loss of sensitive data. Based on the severity level for each of the 37 top vulnerabilities, we can calculate their average threat level at 3.7, somewhere between moderately and highly critical.

Extremely dangerous software flaws

In this section we analyze eight vulnerabilities, selected from 37 software security flaws that are actively used by cybercriminals in widespread exploit packs. Although most of the more commonplace vulnerabilities are found in Adobe products, the most frequently exploited loopholes are actually in Oracle Java.

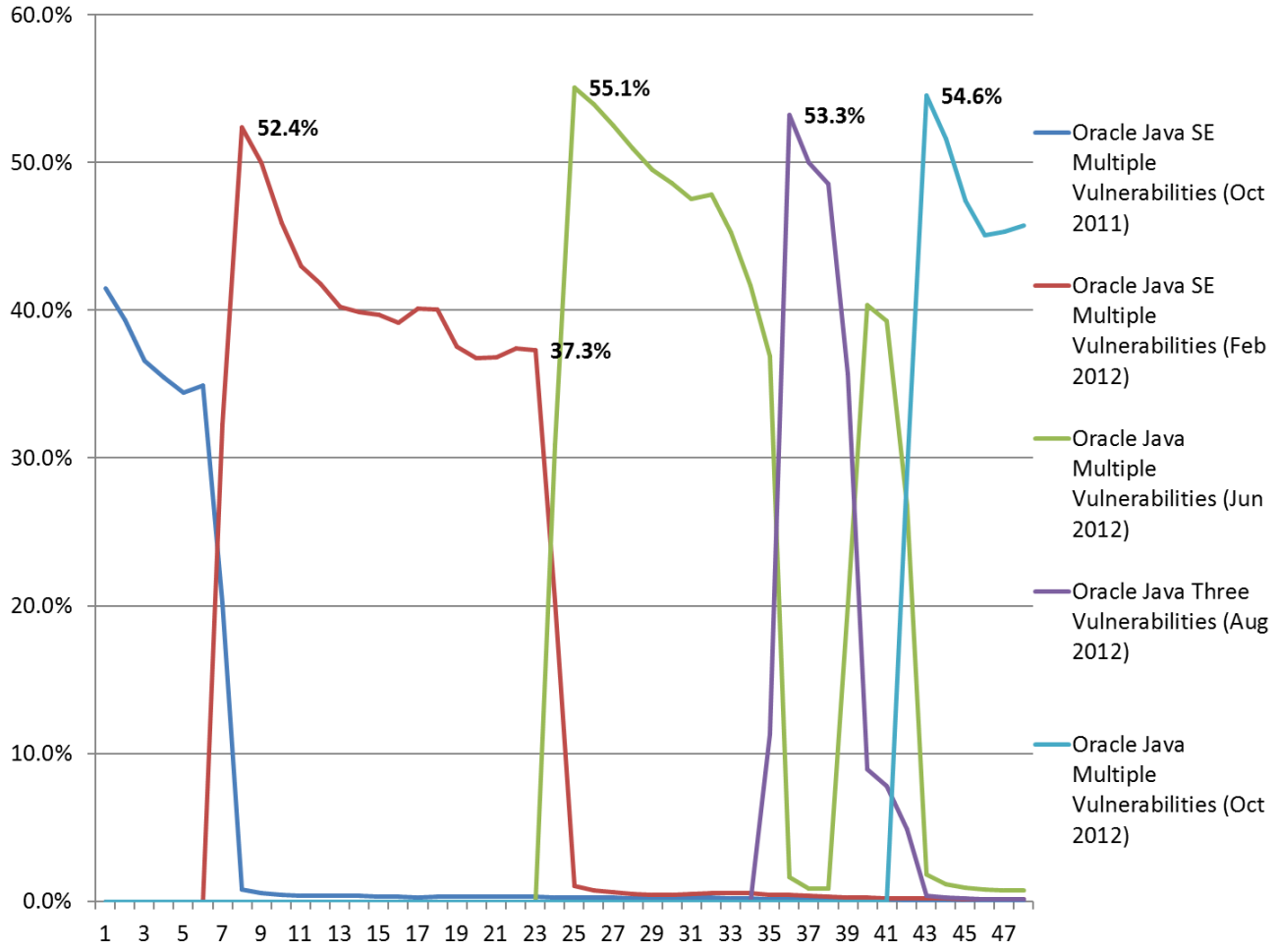


Number of actively exploited vulnerabilities, by software where vulnerability is found

Let's further analyze these vulnerabilities, grouped by the respective software.

Oracle Java vulnerabilities

Java is an obvious “leader” in terms of discovered vulnerabilities, and 2012 was a very tough year for Oracle. We recorded five major vulnerabilities in this software, with the earliest one discovered in October 2011 and the most recent one in October 2012. The evolution of Java vulnerabilities and their prevalence is displayed in this chart:



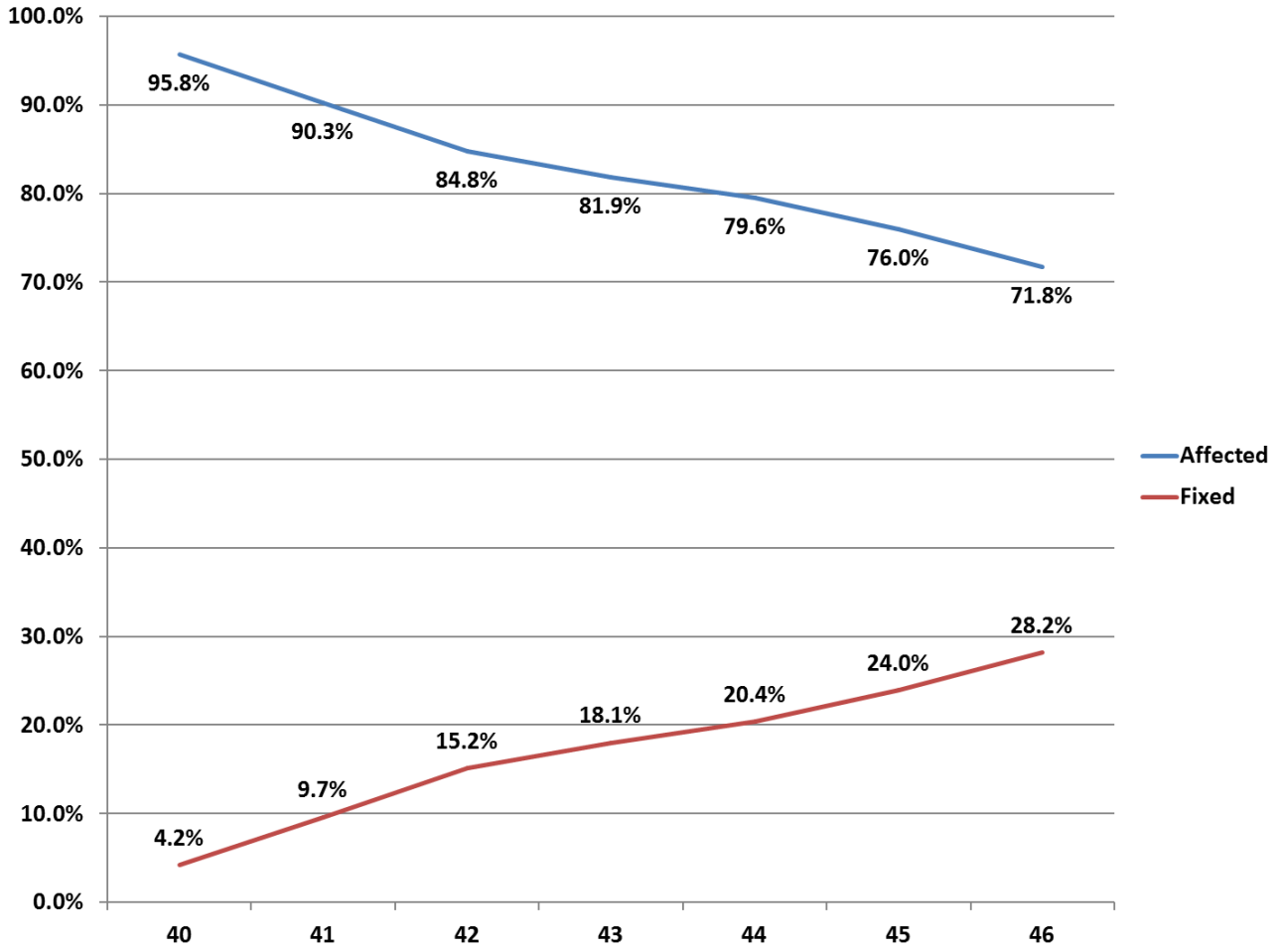
Prevalence of Oracle Java vulnerabilities in 2012

In a vulnerability scan, only one potential weak spot is recorded for each program, even though it might be prone to several security vulnerabilities. However, in the case of Oracle Java, all five of these vulnerabilities were actively exploited by cybercriminals. This means we have to consider all of them to assess how many users are affected. As we can see, at any given time in 2012 there were a large number of users at risk from Java vulnerabilities. At the lowest point, in February, more than one in three (34.5%) were affected; the high water mark came in October when a combination of three vulnerabilities affected 61.1% of users.

We can also see that users are extremely reluctant to switch to the updated software, even when this will fix dangerous security issues. In one particular case with multiple Java vulnerabilities discovered in February 2012, the highest recorded share of affected users was 52.4% at the end of February 2012. The update for Java versions 6 and 7 was released on 14 February. 16 weeks (or four months!) later, it dropped to only 37.3% - still a substantial figure. During this period another update of Java was released (26 April) with non-security fixes, and at the end of it (12 June) one more update came up, fixing newly discovered security flaws. In other words, users had approximately four months to switch to the new version (secure at that time), but it took an astonishingly long time for them to react.

Java Real Usage Analysis

We carried our further analysis into the actual use of Java software in the period between two updates. On 30 August, Oracle launched Java SE 7 Update 7 and Java SE 6 Update 35. 16 October saw the arrival of Java SE 7 Update 9 and SE 6 Update 37. All these updates covered serious vulnerabilities. Using an alternative source of data from our users, a source which looks at the actual software in use, we discovered 41 different major versions of Java 6 and 7 being used. The vulnerability addressed in the 30 August Update (details can be found here at the Oracle website) also affects all prior versions of Java. Therefore, we combined the share of all previous (Affected) versions and compared them with the two newly updated (Fixed) versions. The results can be observed in this chart:

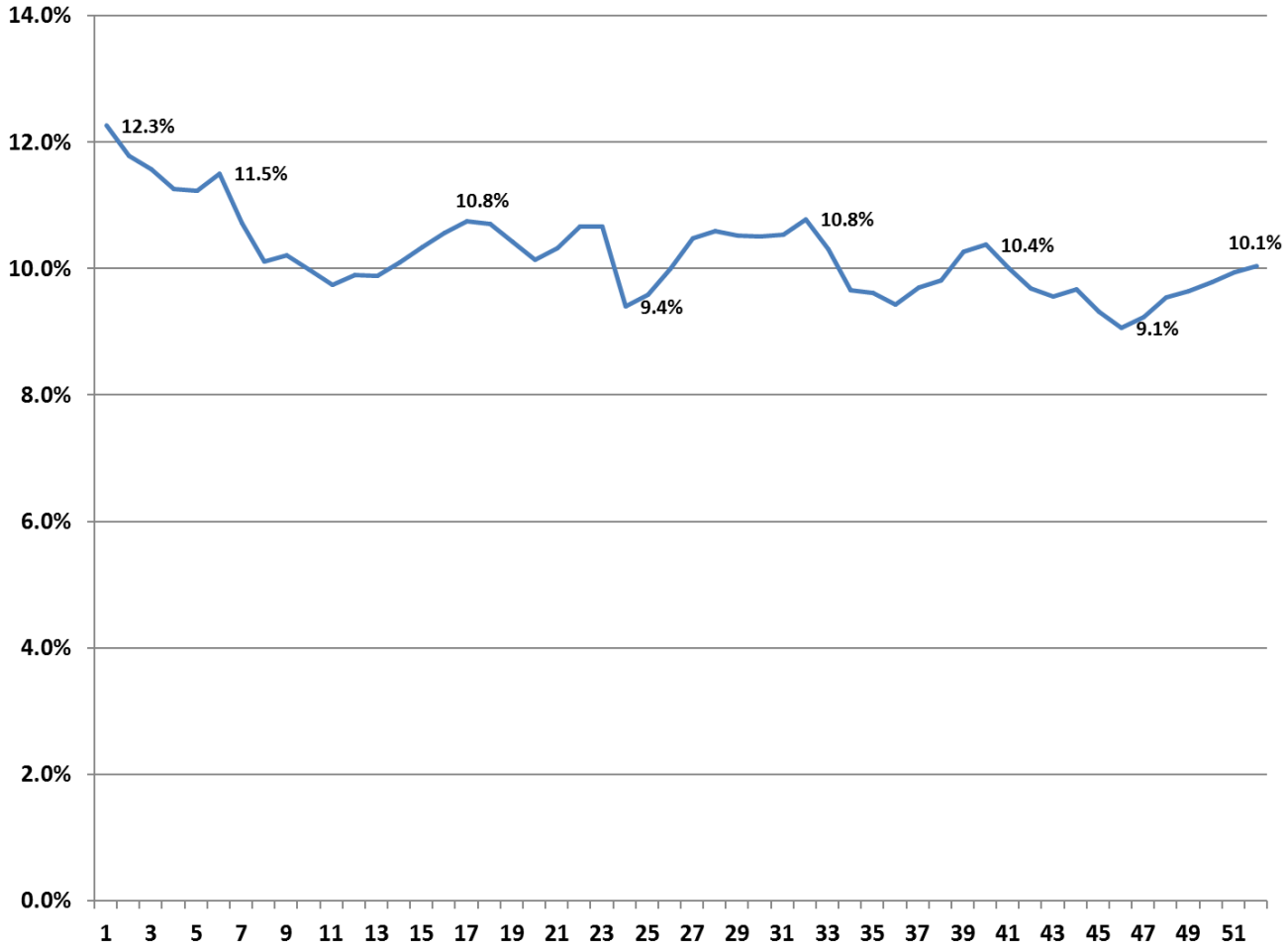


User share of newer versions of Java (Fixed) compared to older and vulnerable versions (Affected), on a weekly basis.

Knowing the high impact of Java vulnerabilities, we used a further method to analyze how fast users switch to the newer version of this software, when faced with an actively exploited vulnerability in the previous one. In this case users had seven weeks to update the secure (at that time) version of Java 6 or 7, but less than 30% of users actually managed to do that, before a newer version (fixing yet another set of multiple vulnerabilities) was released. In a previous report on web browser usage, we used similar data to calculate the upgrade speed for Google Chrome, Firefox and Opera. In all three cases 30% or more of users switched to the newer version within a week after the initial release. Clearly, we can describe the update process for Oracle Java as very slow.

Adobe Flash Player vulnerabilities

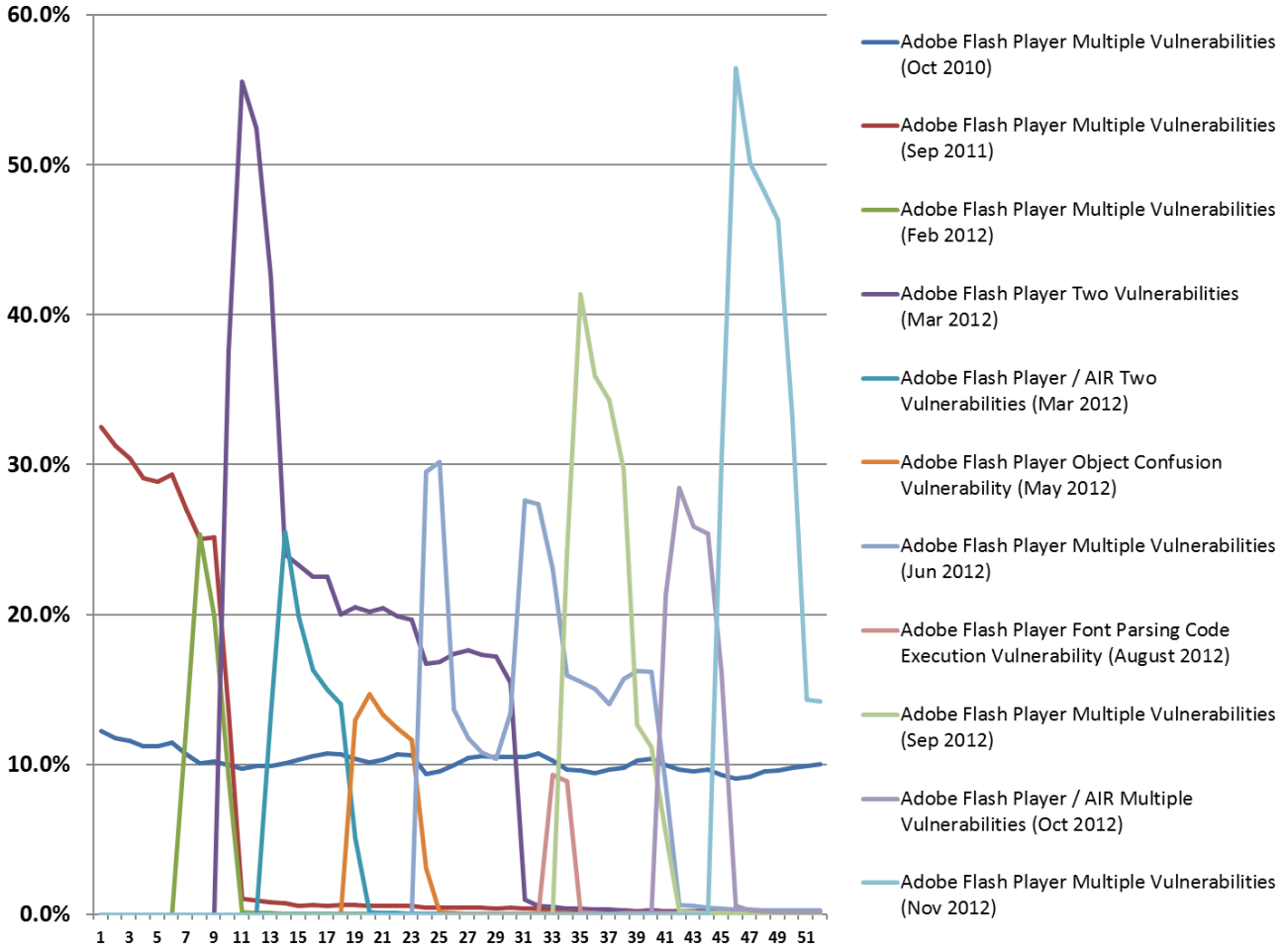
Based on the number of frequently discovered vulnerabilities in 2012, Adobe Flash Player surpasses Java – we detected 11 (!) widespread vulnerabilities during this period (another five came on Shockwave Player, a different type of software). Fortunately, only two of them were in fact exploited by cybercriminals (compared to five for Java). First, we would like to highlight one particular Flash vulnerability that stands out from the crowd.



Relative share of Adobe Flash Player vulnerabilities discovered and fixed in October 2010, on a weekly basis.

Unlike other Adobe Flash Player vulnerabilities that we will analyze later, this one was discovered and fixed more than two years ago. But as we can see, users who have this particular version were not informed about the update or have been reluctant to respond to automatic update notifications. The obsolete and vulnerable Adobe Flash Player was installed on 10.2% of computers on average – an astonishing amount of machines, considering the fact that an exploit was confirmed to exist for this vulnerability, but was not actively used. It seems possible that this vulnerability will only disappear when all computers currently running obsolete software are replaced with new ones.

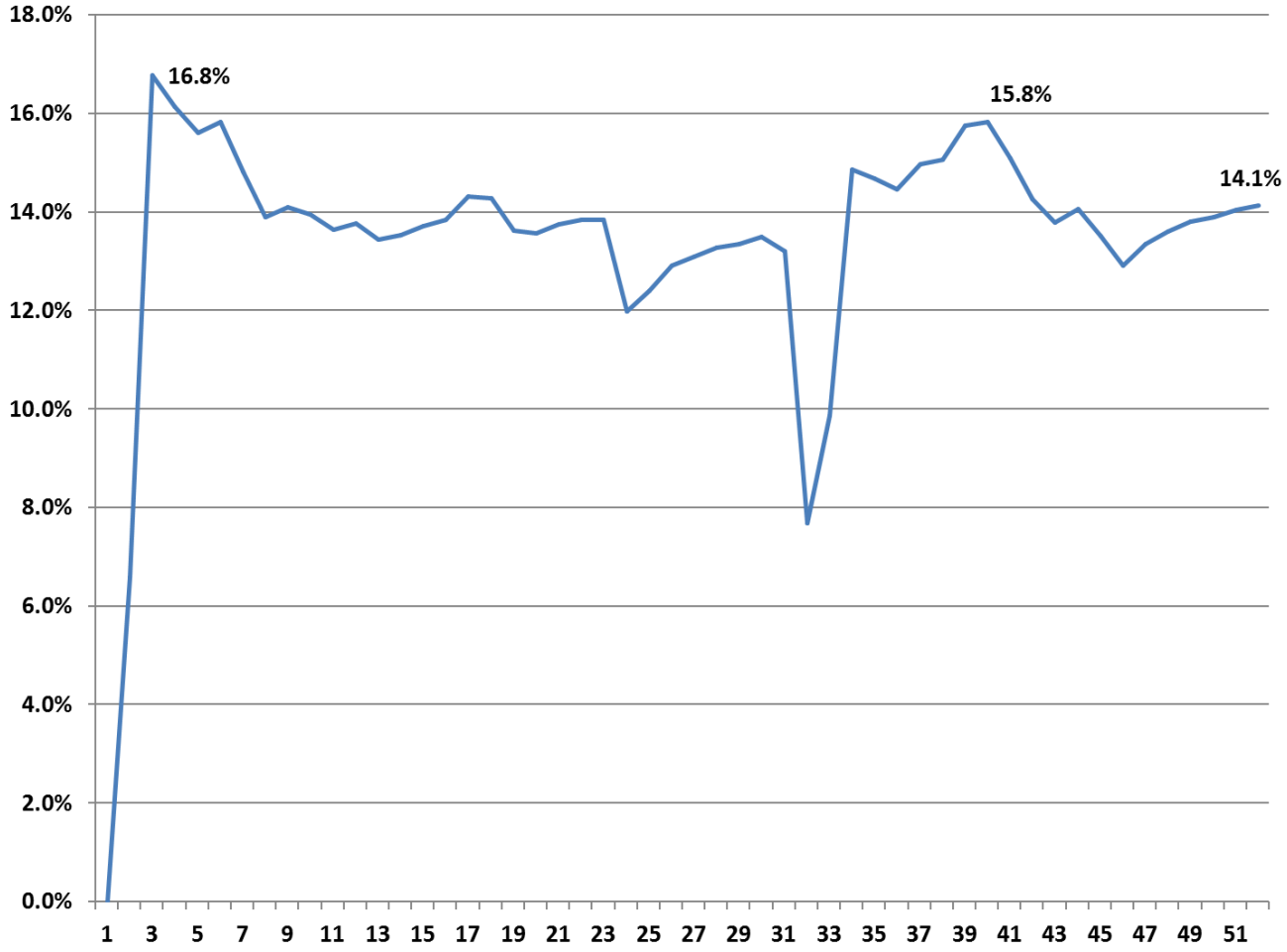
Flash Vulnerabilities: The Big Picture



Relative levels of Adobe Flash Player vulnerabilities in 2012

The picture among the other 10 Flash vulnerabilities is more complex. Again, the vulnerability scan only uncovers one vulnerability per program, which is why this chart shows newer vulnerabilities overlapping some vulnerabilities and replacing others. Even though only two vulnerabilities out of the displayed 10 are actually exploited (the ones discovered in May and August 2012), the chart also shows that there is little sign of the level of vulnerabilities falling away: users are very slow to switch to the newer versions of software, regardless of how dangerous the discovered vulnerabilities might be. In future reports we will focus on actual usage statistics for Adobe Flash Player, to define the exact speed of upgrades from version to version.

Adobe Reader/Acrobat vulnerability



Change in a vulnerability's share, on a weekly basis

The only Adobe Reader vulnerability that was popular and actively exploited at this time was discovered in early December 2011, and was found on an average of 13.5% user computers in 2012. Although it was patched immediately, that number did not change much, peaking at 16.8% in January 2012. As with the other programs analyzed, there is little evidence of any decrease in this vulnerability. Once again, it suggests that users are reluctant to update their software – probably because of an inefficient automatic update system.

Conclusion

This research allowed us to look at the threat level of software vulnerabilities from a unique point of view: showing software security flaws as a time bomb waiting to be detonated by a cybercriminal. Our information comes from our users, who are protected by our products and are therefore less likely to fall victim to an exploit. Even allowing for this, though, the picture is not pretty.

Even when a software vendor does its best to recognize a security flaw and releases an update in a timely manner, this means nothing for a significant proportion of users. A known, dangerous and exploitable security hole remains open on millions of PCs months after it was discovered and an update was provided. There are examples of software vulnerabilities that last for years after being discovered and fixed.

We can't really blame users for that: they are not, and shouldn't have to be, security experts. What is needed is a more streamlined and automated update process for all installed software and better security practices from vendors in general. What users have to understand is that the freedom to install any version of any program of their choice requires certain precautions – and the starting point is proper protection from modern threats, including the tools to detect and update vulnerable software.

Recommendations for consumers

- Use security software at all times: having the latest versions of all programs does not protect you from the latest exploits, which utilize zero-day vulnerabilities. Kaspersky Lab offers a new technology designed to detect and block even new and unknown exploits, named Automatic Exploit Prevention.
- Using one computer for several years without re-installing an operating system is a common scenario. If this is the case, perform an inventory check of the installed software. Remove programs you are not using at all. Update the rest to the most recent version, if possible.
- Use special software to check installed programs for vulnerabilities. Do your best to update those programs that are critically vulnerable. The easiest way to do this is to use a full featured security suite like Kaspersky Internet Security.
- When doing this, pay close attention to Oracle Java, Adobe Flash Player and Adobe Reader – these programs are exploited most frequently.
- If you are using an Apple computer, do not assume you are immune from vulnerabilities. Unfortunately, they are often cross-platform. For example, the infamous Flashfake botnet used a vulnerability in Java. The same applies to Linux: although they are less frequently attacked by cybercriminals, they may become a gateway for a targeted attack on a company.

Recommendations for businesses

- The software being used in your company has to be under control. In other words, you have to understand, which programs and which versions are used by employees, and, of course, whether those versions are safe. In general, application control has to be done in a centralized, real-time manner, as offered, for example, by Kaspersky Endpoint Security for Business coupled with the Kaspersky Security Center management console. Its diverse tool set keeps track of applications, devices and web activity.
- When it comes to vulnerabilities, the first step is an inventory check. Indeed this task is often so complex it becomes almost as demanding as protecting against cybercriminal activity for many IT departments. Happily, Kaspersky Lab's new corporate solution allows you to take control over all aspects of IT Security, starting with a hardware and software inventory and including everything up to the easy configuration and deployment of endpoints and other nodes on the network. Vulnerability assessment and versatile patch management functionality are among the highlights of these new features.

- The first vulnerability check may reveal quite a grim picture, since business security policies often limit employees' ability to upgrade a program themselves. As we have shown in this report, many users don't even think about upgrading an application which appears to function normally. Adding limited privileges to this equation may result in obsolete software being used by all employees company-wide for years at a time.
- So what's the solution? First, take a closer look at the most frequently used vulnerable software described in this report. Compare it with programs used in your company. These are the most likely victims of a targeted attack – all of them start with an exploit, aimed specifically at software used by your employees. But also pay attention to "long-standing" vulnerabilities, which have remained widespread over a very long time. Corporate software updates, like the inventory, should be performed in a centralized way using a proper patch management solution.
- An extra way to enhance corporate security is the new Default Deny scenario. In Kaspersky Lab's corporate products it uses a vast 600 million-plus database of legitimate programs to ensure that only known software with a good reputation is allowed to run. Programs that are not allowed by the company, unknown software and malware are fully restricted by this scenario. At the same time, the work of mission-critical programs and system components is ensured by a special "Golden Image" category of essential applications, and Kaspersky Lab's partnership with over 200 major software vendors guarantees that the most recent versions of core software are added to the "allowed" list.
- Encrypting critically important data can reduce the chance of information leakage. One of the main goals of a targeted attack on any company is data theft, and encryption will help to protect your information even if an infection is successful.