Secunia
Stay Secure

Secunia Yearly Report 2011

The evolution of software security from a global enterprise and end-point perspective.

Published February 14th, 2012

Contents

Secunia Yearly Report 2011 1
Executive Summary 3

1. Vulnerabilities Are Resilient 4
A dangerous thread woven through the software industry 5
A brief history of global vulnerabilities (2006-2011) 7
Dissecting the archetypal software industry 9
Are vendors sharing vulnerabilities? 11

2. Software Is Under Attack 12
Billions of potential cybercrime victims 13
The Top-50 software portfolio under the microscope 14
Patching a typical end-point 18
Attacking a typical end-point 19

3. Does Popularity Determine Exploitation? 21
Market share doesn't determine risk level 22
Impacts of market share 23
Exploit availability according to prevalence 23

4. Moving Targets – Knowing What To Patch 24
Dynamic threats require agile, strategic tactics 25
The risks for enterprise portfolios 27

Appendix 28
1. Secunia Vulnerability Tracking Process 29
2. Secunia Vulnerability Criticality Classification 29
3. Attack Vector 30
4. Genuine and Shared Vulnerabilities 30
5. The Top-50 Software Portfolio 31

Glossary 33

Executive Summary

No end-user or organisation would contemplate leaving the front door to their home or office unlocked as their private property and confidential information could be exposed to theft.

However, many are still leaving themselves at risk from another angle. By not addressing vulnerabilities (errors in software installed on end-points that can be exploited), these very same end-users and organisations are effectively leaving their 'windows' wide open as entry points for cybercriminals to compromise sensitive financial/employee/personal data. Indeed, everyone who uses the Internet – 31% of the Earth's population – is a potential victim of cybercrime.

The Secunia Yearly Report 2011[1] therefore focuses on the evolution and threats of software vulnerabilities, software vulnerability exploits, and the challenges involved in protecting private users and corporate infrastructures reliant on information technology.

Analysing data from 2006 to 2011 reveals that the software industry is still unable to reduce the number of vulnerabilities in software. Comparing the average number of vulnerabilities affecting the products of the Top-20 vendors, it is clear that none of these vendors managed to reduce the number of vulnerabilities in their products. Identifying and remediating vulnerabilities in deployed products therefore remains a critical task for organisations and private users in order to manage the risks of security breaches and system compromise.

In the last few years, vulnerabilities affecting typical end-points more than tripled to over 800 and the majority of these (79%) were found in third-party (non-Microsoft) programs. Third-party programs are considerably more difficult to patch as several different update mechanisms are required to do so. Taking the approach of only securing the operating system and Microsoft programs leaves the end-point at considerable risk. However, the power to protect end-points is in the hands of all users as 72% of the vulnerabilities had a patch available on the day of vulnerability disclosure.

Focusing on the sheer number of vulnerabilities is just half of the story. The shifting dynamics of the threat landscape means that knowing what to patch – what programs cybercriminals are setting their sights on – and when, is just as critical. It is a common fallacy that exploits are mostly available for popular programs, such as Microsoft programs. In fact, there can be a significant gap between what an organisation patches vs. what a cybercriminal has the opportunity to, or chooses to attack. Importantly, this analysis reveals that programs with low market share are also at risk.

The simple truth is that exploitation of any program can compromise an entire end-point and cause potential consequences such as financial losses, theft of personal information, extended downtime, data compromise, and damage to brand image and customer confidence. A strategy of patching a limited and statically defined set of programs considered business-critical fails to reduce the true risks due to the dynamic, rapidly changing threat environment. Efficient identification of the truly vulnerable programs, and then patching those first, is therefore the optimal approach to achieve the largest reduction of risk with limited resources.

1: Past reports and related resources are available for download at http://secunia.com/resources

Vulnerabilities Are

Resilient

I

A dangerous thread woven through the software industry

A clear picture of the IT security ecosystem in terms of vulnerabilities in software is revealed by analysing six years of vulnerability intelligence data. These insights confirm that the software industry is in static mode – still unable to reduce the number of vulnerabilities in software. Results such as these undoubtedly have a negative knock-on effect on organisations and private users alike, placing them high on cybercriminals' radars. Therefore identifying and remediating vulnerabilities in deployed software remains a key task in order to manage the associated risks of system exploitation.

One of the most significant changes over the past few decades has been the rise of information technology and security as important, integral parts of everyday financial activities and communication. For example, worldwide Internet usage has grown by 448% to an estimated 2 billion users since 2000, and networking has evolved from dedicated point to point connections to ubiquitous communication between people, platforms, and applications. At the same time, the complexity and variety of software typically found in private and corporate use has generally increased. Vulnerabilities in software continue to be a major contributor to the risks that people, as well as organisations, face when using software and the Internet.

A vulnerability is an error in software which can be exploited with a security impact and gain. As such vulnerabilities are the major attack vector that opens the door for unauthorised system compromise. Cybercriminals motivated by profit, as well as opportunistic attackers, use refined methods to identify and exploit vulnerable systems connected to the Internet in an automated fashion and on a large scale. Furthermore, cybercriminals routinely develop and deploy robust processes to systematically and successfully bypass detection by anti-malware and anti-virus products[2].

Accurate information about vulnerabilities is a major factor in understanding the threats and is thus a prerequisite to understanding how to successfully remediate the risks.

Vulnerability Tracking

Tracking the development of software security is a complex undertaking. Therefore, several metrics, as listed in Table1, are used to provide deeper insights into the evolution of software security over the last few years.

The main source of information behind this analysis is the Secunia Vulnerability Intelligence database; an impressive data-set which contains information about more than 30,000 products and 5,000 vendors. Secunia validates, verifies, and assesses and tests the vulnerability information gathered and included in the database with consistent and standard processes that have been constantly refined over the years. Details about this process are provided in Appendix 1 of this report.

Vulnerabilities in software continue to be a major contributor to the risks that people, as well as organisations, face when using software and the Internet.

2: How to Secure a Moving Target with Limited Resources - http://secunia.com/products/corporate/csi/howtosecure/

Secunia Advisory	The number of Secunia Advisories published in a given period of time is a first order approximation of the number of security events in that period. Security events stand for the number of administrative actions required to keep the specific product secure throughout a given period of time.
Common Vulnerabilities and Exposures (CVE)	Common Vulnerabilities and Exposures (CVE) is a dictionary of publicly known information security vulnerabilities and exposures. CVE has become a de facto industry standard used to uniquely identify vulnerabilities which have achieved wide acceptance in the security industry. Using CVEs as vulnerability identifiers allows correlating information about vulnerabilities between different security products and services. CVE information is assigned in Secunia Advisories. The intention of CVE identifiers is, however, not to provide reliable vulnerability counts, but is instead a very useful, unique identifier for identifying one or more vulnerabilities and correlating them between different sources. The problem in using CVE identifiers for counting vulnerabilities is that CVE abstraction rules may merge vulnerabilities of the same type in the same product versions into a single CVE, resulting in one CVE sometimes covering multiple vulnerabilities. This may result in lower vulnerability counts than expected when basing statistics on the CVE identifiers.
Secunia Vulnerability Count	A vulnerability count is added to each Secunia Advisory to indicate the number of vulnerabilities covered by the Secunia Advisory. Using this count for statistical purposes is more accurate than counting CVE identifiers. Using vulnerability counts is, however, also not ideal as this is assigned per advisory. This means that one advisory may cover multiple products, but multiple advisories may also cover the same vulnerabilities in the same code-base shared across different programs and even different vendors.

Table 1 - Metrics used to count vulnerabilities in software.

Despite the Secunia vulnerability count being a technically more accurate metric, CVE identifiers are used as a representation of the number of vulnerabilities in this report because these can be counted "uniquely" and CVE is the de facto industry standard for correlating different sources.

A brief history of global vulnerabilities (2006-2011)

Analysing the long-term and short-term trends of all products from all vendors in the Secunia database over the last six years reveals that the total number of vulnerabilities decreased slightly in 2011 compared to 2010.

Figure 1 shows that, on average, 3,550 Secunia Advisories, 4,645 CVEs, and 8,663 vulnerabilities were counted in the years from 2006 to 2011. These vulnerabilities subsequently affected, on average, 2,975 products from 568 different vendors in this period.

Figure 1 concludes that the year 2006 still stands out as the all-time high with respect to these metrics. It can be observed that, except for the Secunia vulnerability count, all metrics show a decreasing trend in the long-term (5 years) and short-term (2010 vs. 2011) of between 5% and 24%. It should be noted that analysing vulnerability counts covering all products includes a large number of rare products and web applications that are not in typical everyday use in organisations or on private systems. However, from this high-level perspective it is clear that, globally, the exponential growth in vulnerability numbers observed up to 2006 has essentially flattened.

This is highlighted in Table 2, which shows the downward trend in Secunia Advisory and CVE counts with fluctuations of around 15% for the last five years.

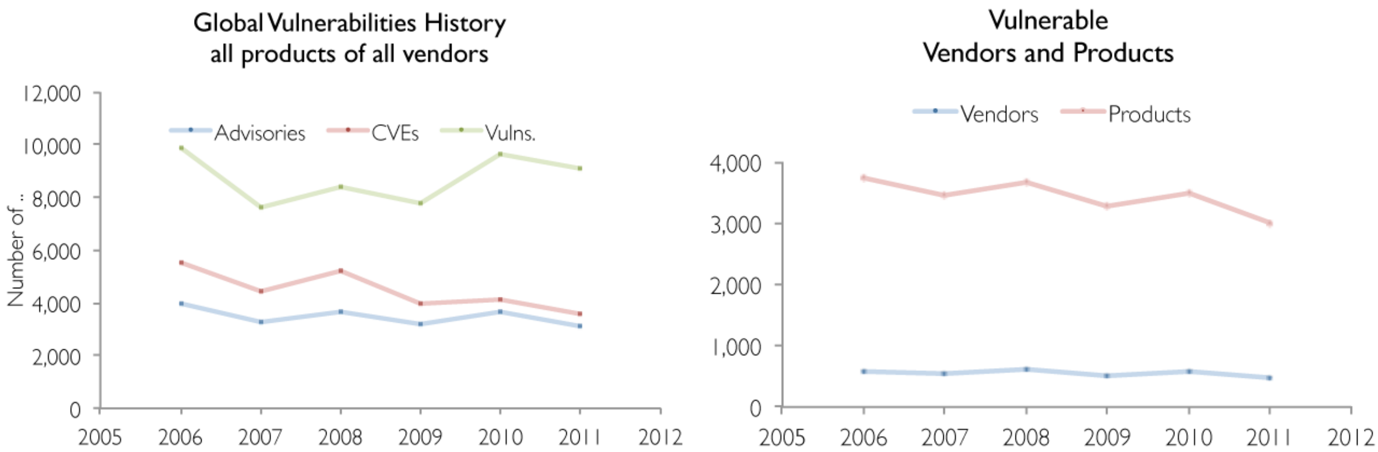


Figure 1 – A history of vulnerability counts and the number of vendors and products.

	Secunia Advisories	CVEs	Vulnerability count	Vendors	Products
Average 2006-10	3,550	4,645	8,663	568	2,975
Total 2011	3,111	3,551	9,132	477	2,527
Trend 5 years	-12%	-24%	5%	-16%	-15%
Trend 2010/11	-15%	-14%	-5%	-19%	-16%

Table 2 – The average number of vulnerabilities, trends, and the number of unique vendors and products affected by these vulnerabilities.

Criticality and attack vector

Figure 2 displays the criticality rating[3] and attack vector[4] for all Secunia Advisories released in 2011. More than half of the vulnerabilities in 2011 were rated as "Medium", "Highly", or "Extremely critical".

The prevalence of the medium- to high-level criticality ratings in combination with the attack vector, as reported in Figure 2, clearly shows that the majority of these vulnerabilities represent a genuine threat with an increased risk of system compromise. Significantly, most of the vulnerabilities are exploitable from a remote network and nearly 20% of the vulnerabilities are rated as "Highly" or "Extremely critical". This demonstrates that the majority of the vulnerabilities are relevant and require urgent, dedicated attention from a risk management perspective, particularly as attackers only need a single exploitable vulnerability to compromise the entire end-point.

While the observed high-level global trend of decreasing vulnerability counts (all products from all vendors) is encouraging, it should be noted that in absolute terms, the numbers remain considerably large. These high vulnerability counts, paired with the high criticality ratings, indicate that accurate information about vulnerabilities is an essential, security-critical requirement for effective risk assessment, prioritization, and vulnerability remediation.

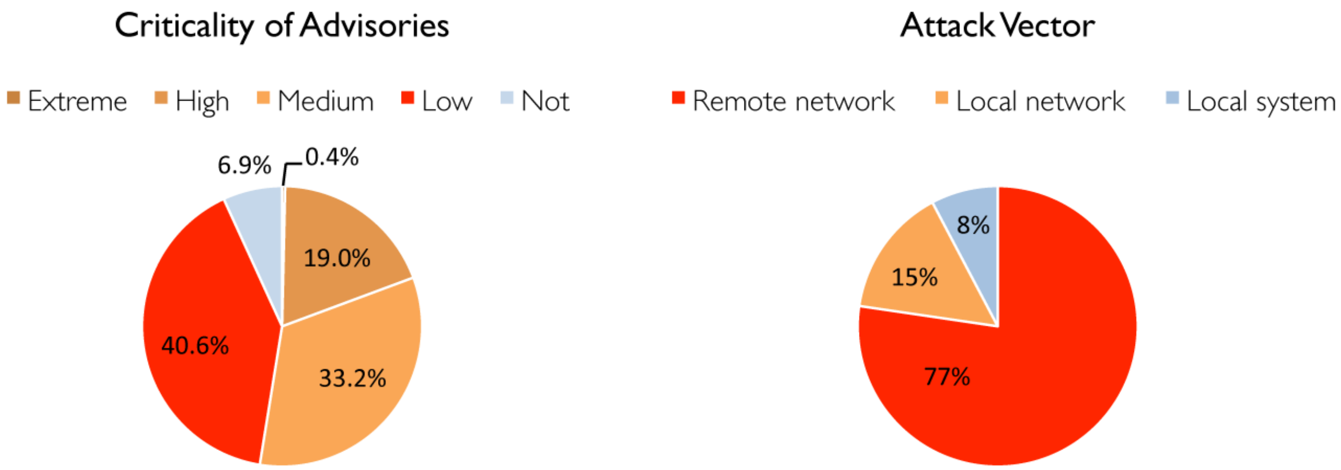


Figure 2 – Distribution of criticality and attack vector location for 2011.

The majority of the vulnerabilities are relevant and require urgent, dedicated attention from a risk management perspective.

3: The criticality of vulnerabilities is rated on a five-level criticality scale. Classifications and descriptions of how they are used to rate the risk of a vulnerability are listed in Appendix 2 of this report.

4: The attack vector describes the way an attacker can trigger or reach the vulnerability in a product. Classifications and descriptions of how they are used in Secunia Advisories are listed in Appendix 3 of this report.

Dissecting the archetypal software industry

Counting all of the vulnerabilities for all of the products from all of the vendors, as it is usually exhaustively done at the end of a year by various organisations, only provides a superficial picture. Such an analysis naturally includes a large number of rare and special products, or an increasing number of web application vulnerabilities that only affect a small part of a user's or an organisation's operating software.

Therefore, in order to gain a better insight into the evolution and security of the software industry, a representative sample set covering the most important types of software that make the Internet run must also be investigated.

The Top-20 vendors

To represent and track the evolution of the software industry, the Top-20[5] producers of the software (commercial or open source) with the most vulnerabilities discovered in their products in 2011, were selected.

This representative set of Top-20 vendors comprises:

- Major and diverse producers of software (commercial and open source)
- Major operating systems (different flavours of Linux, Microsoft Windows, Mac OS, Solaris, IBM, and HP-Unix)
- Major browsers (Microsoft Internet Explorer, Google Chrome, Mozilla Firefox, Apple Safari, and Opera)
- Major office products (Microsoft Office, Open Office, and Apple iWork)
- Major networking gear (Cisco)
- Major databases (Oracle, DB2, MySQL, and MS-SQL)
- Major web servers (Apache and MS IIS)

Evolution, risk distribution, and trends

Whatever software private users or organisations run, it is certain that they have programs deployed in their software portfolio from several of these Top-20 vendors.

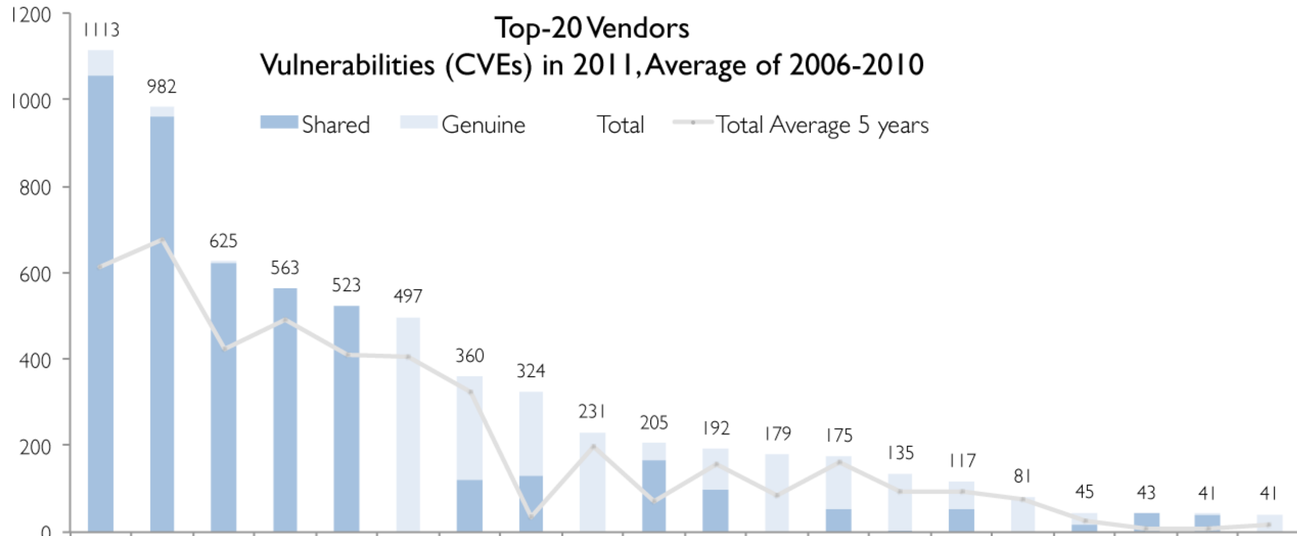
#	Vendor	History 2006-11	2011 CVEs	Risk	Trend 5yr		1yr	
1	Novell		1,113		+81%	▲	+32%	▲
2	Red Hat		982		+45%	▲	-5%	▼
3	Canonical		625		+48%	▲	+9%	▲
4	Debian		563		+15%	▲	+33%	▲
5	Gentoo		523		+28%	▲	+154%	▲
6	Oracle		497		+27%	▲	+34%	▲
7	Apple		360		+12%	▲	-17%	▼
8	Google		324		+800%	▲	+116%	▲
9	Microsoft		231		+17%	▲	-20%	▼
10	VMware		205		+193%	▲	+63%	▲
11	IBM		192		+21%	▲	-19%	▼
12	Adobe		179		+106%	▲	-16%	▼
13	HP		175		+9%	▲	-34%	▼
14	Cisco		135		+41%	▲	+7%	▲
15	Mozilla		117		+26%	▲	+2%	▲
16	Kernel		81		+8%	▲	-21%	▼
17	Apache		45		+88%	▲	+18%	▲
18	Xerox		43		+330%	▲	+2050%	▲
19	Attachmate		41		+583%	▲	+273%	▲
20	Opera		41		+116%	▲	+28%	▲

Figure 3 – Evolution, risk distribution, and trends in vulnerabilities for the Top-20 vendors. Ranking is not according to security level.

Figure 3 presents the long-term and short-term trends; comparing the average number of vulnerabilities of the previous five years (2006 to 2010) to the 2011 numbers, and the 2010 numbers to the 2011 numbers respectively.

As illustrated, the last few vendors listed have considerably lower and very similar or equal numbers of vulnerability counts (~ 4% of the top vendors), which justifies the cut-off at 20 vendors. Combined, the products of these Top-20 vendors were affected by 2,227 unique vulnerabilities (CVEs) in 2011, representing 63% of all vulnerabilities discovered in 2011. This further indicates that the Top-20 vendors represent a relevant sample of the software industry.

5: The following vendors were merged to reflect recent acquisitions in the software industry: Novel also includes SUSE Linux and Oracle also includes Sun Microsystems and BEA systems.



	Novell	Red Hat	Canon-ical	Debian	Gentoo	Oracle	Apple	Google	Micro-soft	VM-ware	IBM	Adobe	HP	Cisco	Mozilla	Kernel	Apache	Xerox	Attach-mate	Opera
Total	1113	982	625	563	523	497	360	324	231	205	192	179	175	135	117	81	45	43	41	41
Genuine	56	21	2	0	0	497	240	194	230	39	93	179	121	131	63	81	27	0	1	40
Shared	1057	961	623	563	523	0	120	130	1	166	99	0	54	4	54	0	18	43	40	1
Total Average 5 years	614	676	423	491	410	405	322	36	198	70	159	87	161	96	93	75	24	10	6	19

Figure 4 – The Top-20 vendors/organisations with the most vulnerabilities in their products in 2011, including a breakdown of genuine and shared vulnerabilities and comparison with the average number of the previous five years (2006-2010). The numbers on the top of the bars indicate the number of vulnerabilities (CVE) disclosed in the products of the specified vendors in 2011.

It is important to note that the purpose of this analysis is not to rank the vendors by security; rather it tracks the evolution of a major part of the IT security industry.

Undiminished, accelerated threats

The number of vulnerabilities increased compared to the average of the previous five years. This continued discovery of numerous vulnerabilities in software that is used by private and corporate users every day has a major impact on the security of all systems.

While significant advances in communication, processes, and technology were made by the software industry in the last decade in terms of security, it can still be observed that none of the Top-20 vendors managed to decrease the number of vulnerabilities discovered in their products in 2011 compared to the average of the previous five years. This is clearly evident in both Figure 3 and Figure 4, where the line representing the five year average (2006-2010) remains consistently below the top of the bars representing vulnerabilities discovered in 2011.

While it is encouraging to see that seven of the Top-20 vendors managed to decrease the number of vulnerabilities discovered between 2010 and 2011 (Figure 3), it is too early to confirm a general trend in the industry.

Are vendors sharing vulnerabilities?

The sum of the vendors' vulnerabilities, as indicated in Figure 3 and Figure 4, is larger than 2,227 (the number of unique CVEs for the Top-20 vendors in 2011) as many vendors share products, code, or common libraries. To assess the extent of shared code (measured through shared vulnerabilities) the bars in Figure 4 are split to indicate the number of genuine and shared vulnerabilities[6] per vendor. This reflects the two major software development and distribution models found in the industry. Genuine vulnerabilities affect products that the vendor produces exclusively, while shared vulnerabilities also affect the products of other vendors.

Many products (especially in the open source community) are shared freely and are based upon common software libraries. For example, Linux distributions are a sample of a large collection of open source programs and libraries that are assembled, tailored, and distributed as a bundled product. Many such open source products are also used by commercial vendors; for instance Apple's Mac operating system 10 (Mac OS-X), which is based on FreeBSD and therefore contains a lot of open source components. Similarly, Google's web browser Chrome includes customised binaries from Adobe to integrate the popular Adobe Flash player functionality in their browser. This allows Google to update the Flash player with Chrome's silent update process in order to protect its users.

In contrast to this, genuine vulnerabilities affect products or software that are not shared with other vendors and are therefore specific to one vendor. The Microsoft operating system, for example, is unique to Microsoft as it is developed by Microsoft and is not shared, bundled, or made available as open source.

While a specific vendor is not the originator of vulnerabilities found in shared code, by bundling it within its products the vendor takes ownership and responsibility for handling and patching these vulnerabilities. It is thus the vendor's responsibility to integrate patches and fixes available for these libraries as soon as possible – a good example is Google Chrome patching the integrated Adobe Flash plugin with its silent update process.

Any given software vendor adopts a specific methodology and philosophy regarding how to develop, test, maintain, and distribute software. Some of these Top-20 vendors invest heavily in the security of their products, trying to prevent vulnerabilities, deploying features to make exploitation of existing vulnerabilities more difficult, and setting up proper processes to identify, handle, and patch newly discovered vulnerabilities.

However, despite massive security investments by the industry, vulnerabilities are still rising and increasing manifold. It seems that the industry's security improvements are either nullified by the ever increasing complexity of their products, or that the current state of the industry represents an economic equilibrium between security investments by vendors and the level of insecurity that is acceptable in the market.

As a result of this, there is a continued need for private and corporate users of software to properly handle vulnerability information and remediation in order to manage and reduce the associated risks.

6: Classifications of genuine and shared vulnerabilities, and the total number of vulnerabilities are listed in Appendix 4 of this report.

Software Is Under Attack

2

Billions of potential cybercrime victims

Over the last few years vulnerabilities affecting typical end-points more than tripled to over 800 – the majority of these (79%) were found in third-party (non-Microsoft) programs. Third-party programs are considerably more difficult to patch as several different update mechanisms are required to do so. Only securing the operating system (OS) and Microsoft programs leaves end-points at considerable risk. However, the power to protect end-points is in the hands of all users as 72% of the vulnerabilities had a patch available on the day of vulnerability disclosure.

It is estimated that, today, more than 2 billion users have access to the Internet. This equates to approximately 31% of the Earth's population[7] . With such a high number of potential victims, it becomes clear that end-points have become a primary target for cybercriminals. Even a very low chance of a successful attack can potentially compromise a large number of end-points and turn them into botnets controlled by cybercriminals. The reasons why end-points have become increasingly rewarding targets for attackers are because:

- **End-points are valuable**
  End-points are where the most valuable data is found to be the least protected. By definition, end-points are the access points to all business-critical data, and are therefore lucrative targets for cybercriminals.

- **Everyone is a target**
  Every end-point represents a profitable target for cybercriminals, even if no sensitive data is present. The end-point's computing power and bandwidth provide crucial resources; for example as an infection point, proxy, or for distributed password cracking services.

- **End-points are difficult to secure**
  End-points are extremely dynamic environments with numerous programs and plug-ins installed. Paired with unpredictable usage patterns, this makes them targets that are difficult to defend.

Every end-point represents a profitable target for cybercriminals, even if no sensitive data is present.

7: Source: http://www.internetworldstats.com

The Top-50 software portfolio under the microscope

To assess the exposure of end-points, the number and types of products typically found installed were determined by analysing anonymous data gathered from the 2011 scans of more than 4.6 million registered users of the Secunia Personal Software Inspector (PSI)[8].

Figure 5 visualises the diversity of the users' software portfolios. 50% of users were found to have more than 66 programs installed from more than 22 different vendors. The majority of users (indicated by the area within the plot) lies well outside of this range in terms of vendors and programs, which further demonstrates the large diversity of software found deployed in the field.

To track the security of a typical user in light of this diversity of software portfolios, a representative Top-50 portfolio comprising the most prevalent products found by the Secunia PSI was built. The Top-50 software portfolio, as of December 2011, contains software from 12 different vendors; namely 28 programs from Microsoft and 22 programs from third-parties (non-Microsoft vendors) as indicated in Figure 6.

The resulting Top-50 software[9] portfolio was then correlated with Secunia's Vulnerability Intelligence database to determine the evolution of vulnerabilities affecting these systems, and assess the challenges associated with keeping these end-points secure.

A representative end-point configuration comprising the Windows operating system (Windows XP, Windows Vista, and Windows 7) and a software portfolio with the Top-50 most prevalent programs was also tracked.

To track the security of a typical user in light of this diversity of software portfolios, a representative Top-50 portfolio comprising the most prevalent products found by the Secunia PSI was built.

8: The Secunia PSI is a free, lightweight scanner that identifies and patches insecure programs on end-points - http://secunia.com/PSI

9: The Top-50 software portfolio comprises the most prevalent programs as determined by scans of the Secunia PSI conducted in December 2011. The programs together with market share, number of vulnerabilities, and exploit availability are listed in Appendix 5 of this report.

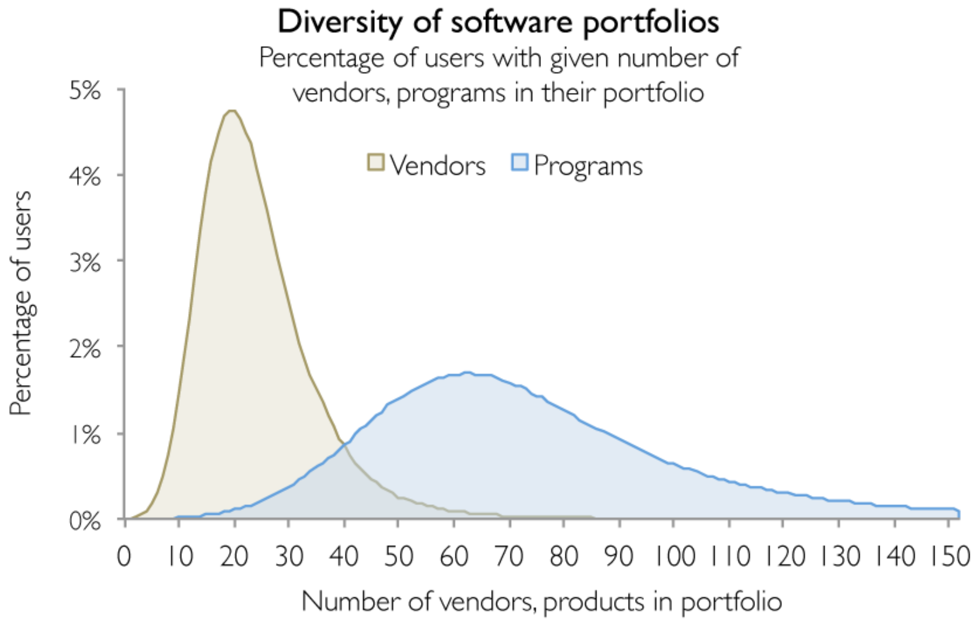


Figure 5 – Diversity of software portfolios. Percentage of users with a given number of vendors (programs) in their software portfolio.

In order to provide the longest history and identify the long-term trends when tracking the security of a typical end-point, the evolution of the Top-50 software portfolio with Windows XP as the operating system is plotted in the left pane of Figure 7.

Additionally tracking the vulnerabilities of the operating systems Windows XP (released in 2001), Windows Vista (released in 2007), and Windows 7 (released in late 2009) reveals that even though Windows 7 is currently the most prevalent operating system on end-points, the choice of operating system has only a minor impact on the total number of vulnerabilities on a typical end-point (Figure 7 and Table 3).

Figure 7 plots the total number of vulnerabilities of the end-point (Total), together with a breakdown by origin of the vulnerabilities.

Analysing the number of vulnerabilities affecting a typical end-point together with the operating system highlighted an alarming trend. The number of vulnerabilities found in the Top-50 software portfolio actually increased more than three-fold since 2007 to 870 in 2011.

The evolution shown in Figure 7 also clearly demonstrates that this increase is almost exclusively due to vulnerabilities found in programs from third-party vendors. These results clearly indicate that end-points have become increasingly exposed targets for cybercriminals – with hundreds of vulnerabilities that, when left unpatched, potentially allow systems to be compromised.

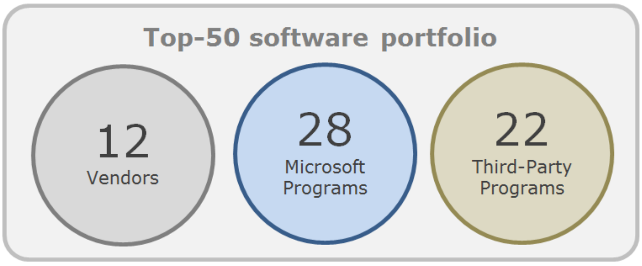


Figure 6 – The Top-50 software portfolio consists of programs from 12 different vendors as of December 2011.

	WinXP	WinVista	Win7
Operating System	101	98	100
Microsoft Programs	84	84	84
Third-Party Programs	685	685	685
Total	**870**	**867**	**869**

Table 3 – Breakdown of vulnerabilities in the Top-50 software portfolio by origin.

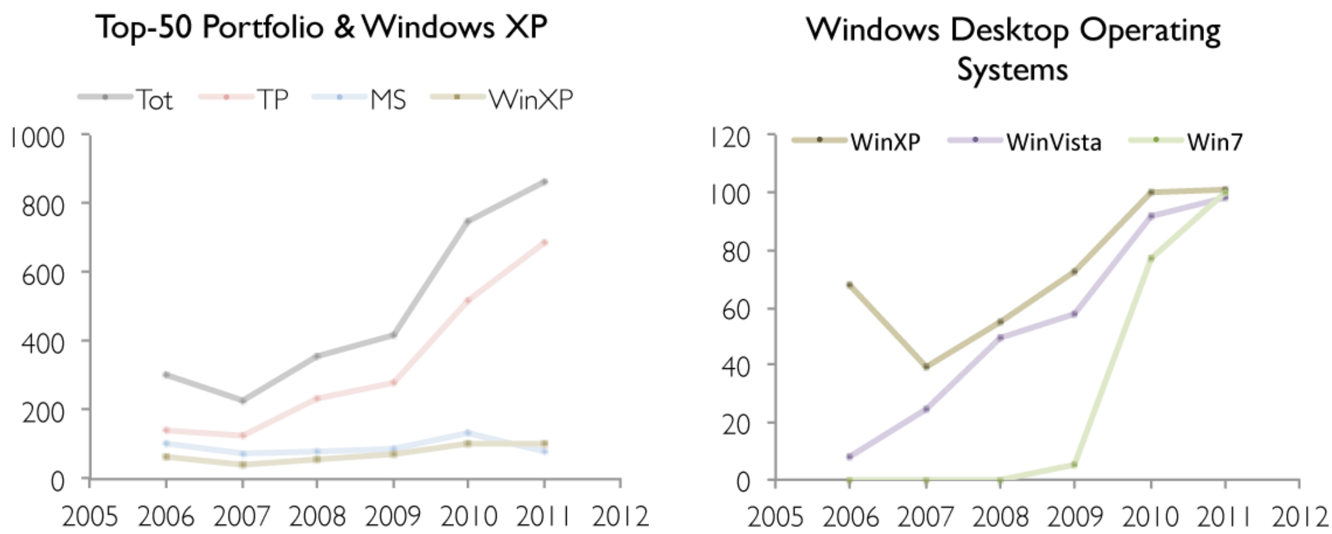


Figure 7 – Evolution of vulnerabilities (CVEs) of a typical end-point with a breakdown according to third-party programs, Microsoft programs, and Windows XP (left). Evolution of vulnerabilities (CVEs) of the operating systems Windows XP, Windows Vista, and Windows 7 (right).

Furthermore, the fact that over the last six years more than 50% of these vulnerabilities were rated as "Highly" or "Extremely critical", as shown in Figure 8, confirms the relevance and importance of this trend. "Highly" and "Extremely critical" vulnerabilities indicate exploitable vulnerabilities that can lead to system compromise where successful exploitation does not normally require any unusual interaction from the user.

Figure 9 confirms that in 2011, 78% of the vulnerabilities affected third-party programs (TP); far outnumbering the 12% of vulnerabilities in the operating system (OS) or the 10% of the vulnerabilities in the Microsoft programs (MS). Significantly, the share of vulnerabilities in third-party programs continuously increased from 45% in 2006 to 78% in 2011.

These statistics represent a worrying trend as the majority of users and organisations still only focus on patching Microsoft programs.

The incorrect perception that Microsoft programs still represent the primary attack vector, means that defences based on this false assumption are as effective as locking the front door to your home while the back door remains wide open.

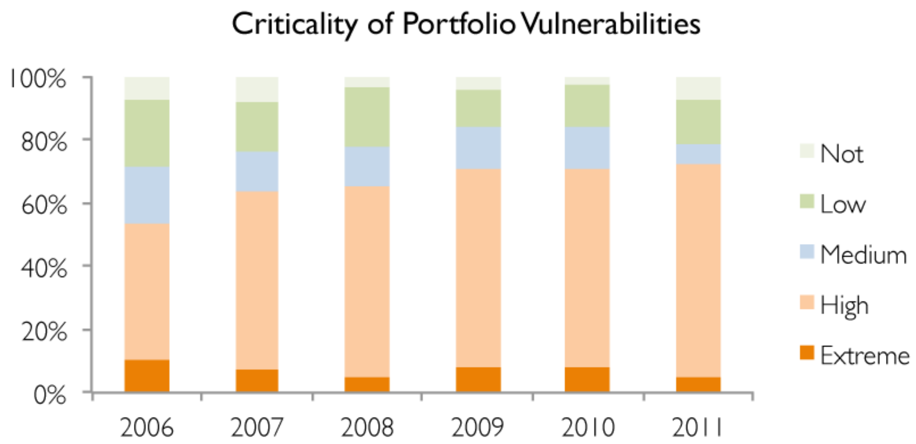


Figure 8 – Evolution of the criticality of the Secunia Advisories of the Top-50 portfolio with Windows XP.

Significantly, the share of vulnerabilities in third-party programs continuously increased from 45% in 2006 to 78% in 2011.

Patching a typical end-point

To fully patch a typical end-point, the user (or administrator of the system) has to master at least 12 different update mechanisms, as the Top-50 software portfolio comprises programs from 12 different vendors, as shown in Figure 6. With one update mechanism, namely "Microsoft Update", the operating system and the 28 Microsoft programs can be patched to remediate 22% of the vulnerabilities.

In addition to this, another 11 update mechanisms are needed to patch the remaining 22 third-party programs to remediate 78% of the vulnerabilities.

This complexity to stay secure will undoubtedly leave a large number of systems incompletely patched – and thus vulnerable to attack and compromise.

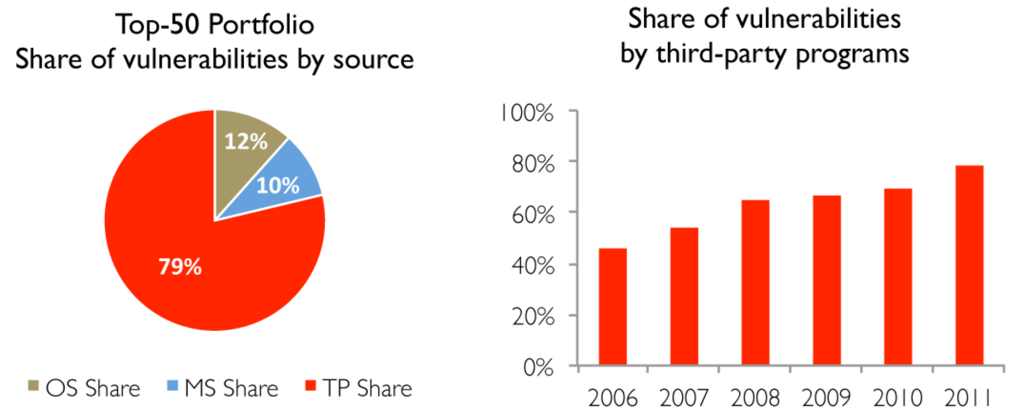


Figure 9 – Share of vulnerabilities in the Top-50 software portfolio by source of programs, including a breakdown of vulnerabilities by origin (left) and the evolution of the share of third-party program vulnerabilities (right).

This complexity to stay secure will undoubtedly leave a large number of systems incompletely patched – and thus vulnerable to attack and compromise.

Attacking a typical end-point

A 0-day exploit is a vulnerability that is exploited before details about the vulnerability is available. The limited feasible protection against 0-day exploits, paired with extensive media coverage, often leads to an overreaction of this threat.

However, the left pane of Figure 10 indicates that for all Secunia Advisories affecting a typical end-point in 2011, 72% had a patch available within one day of the disclosure of the vulnerability, and 77% of the advisories had a patch available within 30 days of disclosure. This data indicates that there is limited room for 0-day

exploits. The 28% of the advisories that had no patch available on the day of disclosure indicates an upper bound of potential for 0-day exploit availability. Microsoft even reports that less than 1% of the attacks in the first half of 2011 were attributed to 0-day exploits[10].

Therefore, the mere possibility of 0-day exploits, a force majeure, does not justify ignoring 72% of the cases where effective remediation is possible and at users' fingertips. Thus, organisations can hardly hide behind the threat of 0-days when a solution is available for 72% of vulnerabilities.

Furthermore, exploit material is available for most of the programs. It is a common fallacy that rare programs (programs with a low market share) are less exposed. Research contradicts this misconception by confirming that even programs with a low market share are exposed to potential compromise as they are not only affected by vulnerabilities, but also exploit material.

Cybercriminals know that the availability of a patch does not imply that the patch is installed in a timely fashion. Thus, the complexity of the task to stay secure is expected to have a direct impact on the security level of end-points. The right pane of Figure 10 indeed confirms that third-party programs are consistently at a lower patch level than Microsoft programs.

10: Microsoft Security Intelligence Report SIR 11 - http://www.microsoft.com/security/sir/

Figure 10 measures the average patch level of all scans for every Sunday in the last 12 months. Sunday was chosen as the sample day as, typically, no patches are released during a weekend.

This approach provides the fairest view as patch adoption is generally at its highest level on the day of patch release. Averaged over a year, 2.7% of the Microsoft programs are found insecure compared to 6.5% of the third-party programs. Thus, on average, more than twice as many third-party programs are found unpatched than Microsoft programs. Figure 10 (right pane) clearly demonstrates that the complexity of the task to keep an end-point secure or fully patched has a direct and measurable impact on security levels. Without a tool or process in place, it remains a difficult task and challenge to stay fully patched over extended periods. However, these findings also demonstrate that cybercriminals have lots of opportunities by exploiting unpatched programs without the need to invest in 0-day exploit material. This measurement is based on a population of Secunia PSI users. Generally, users without the Secunia PSI have a lower patch level.

Figure 10 (left pane) demonstrates the good news: a general patch availability level of 72% on the day of vulnerability disclosure clearly indicates that users are in control of their security for the majority of vulnerabilities. Remediation of the root cause is readily available for private and corporate users to take advantage of.

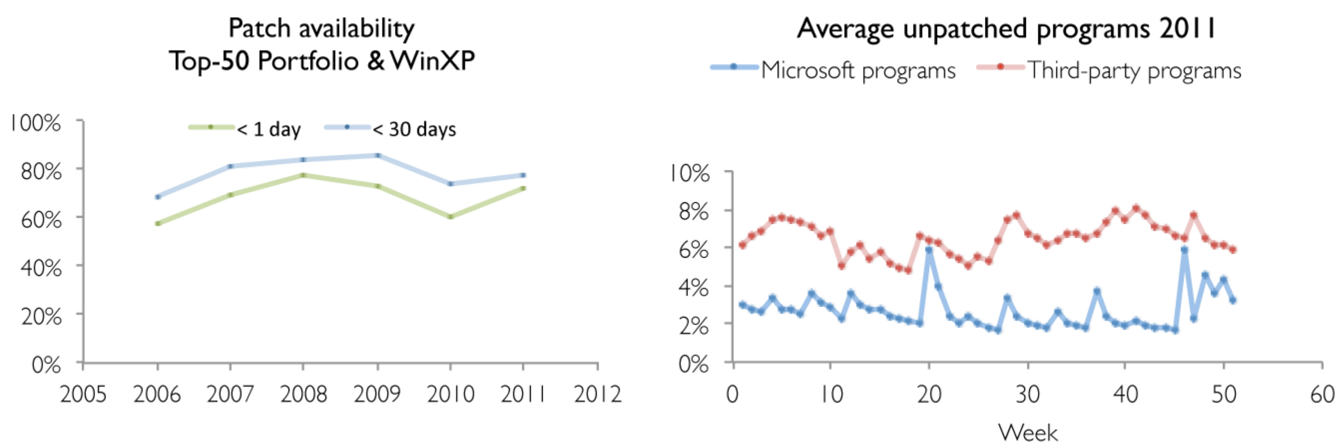


Figure 10 – Percentage of vulnerabilities having a patch available within 1 day or 30 days following disclosure.

Cybercriminals know that the availability of a patch does not imply that the patch is installed in a timely fashion. Thus, the complexity of the task to stay secure is expected to have a direct impact on the security level of end-points.

Does Popularity Determine

Exploitation?

3

Market share doesn't determine risk level

It is a common perception that exploits are mostly available for popular programs. However, this analysis reveals that programs with low market share are also at risk. Thus, patching only a few popular or preselected programs is a highly flawed approach that can leave systems at considerable risk.

The number of vulnerabilities and exploits available are critical factors in determining the risk of a specific product. A vulnerability in a product for which no exploit material is available is a low risk to an organisation. However, if exploit material is available then the vulnerability can pose a serious threat. Most often, exploitation of the vulnerability becomes rather trivial and can be highly automated. Modern malware tools incorporate automated exploitation functionality[11]. Such malware tools are readily and easily rearmed with new exploits through a plugin mechanism. Thus, the unpredictability of exploit availability and the easy integration of new exploit material into attack tools pose a significant threat to all vulnerable products.

It is commonly understood that the higher a product's market share, the more likely it is to have exploits available or contain vulnerabilities. To test this assumption, information about vulnerabilities and exploits was correlated with the market share of the affected products, as measured by the Secunia PSI.

Information about exploit material was compiled from the Secunia Vulnerability Intelligence database and publicly accessible exploit archives. Not all information about exploit material can be automatically correlated to a specific vulnerability or product; therefore the results presented in Figure 11 represent a minimum estimate.

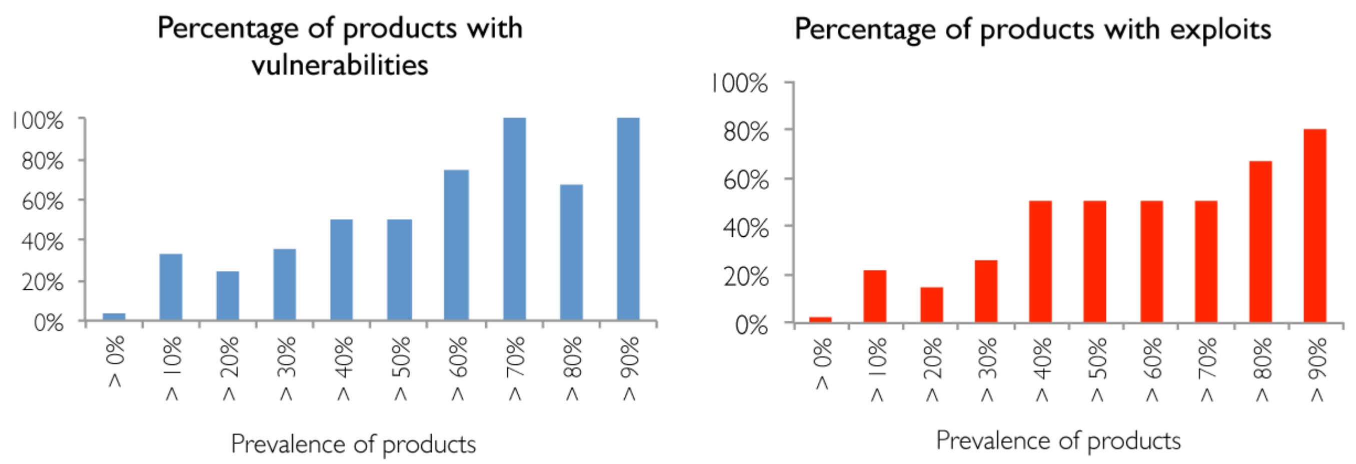


Figure 11 – Percentage of products with a given market share for which vulnerabilities exist (left) or exploit material is available (right) in the last two years.

11: Exploit Packs Run on Java Juice - http://krebsonsecurity.com/tag/exploit-pack/

Impacts of market share

As shown in Figure 11 (left pane), 100% of the products with a market share of between 90% and 100% (indicated as >90% in the plot) had vulnerabilities in the last two years. A considerable percentage of the products with lower market shares also contained vulnerabilities – 33% of the 23 products with a market share of between 10% and 20% (>10%) contained vulnerabilities – which could therefore expose networks, infrastructures, and end-points to potential compromise.

Exploit availability according to prevalence

Analysing exploits confirms that exploit availability positively correlates with the market share of the affected products. The higher the market share, the larger the percentage of products for which exploits are available. Figure 11 (right pane) indicates that for 80% of the products with a market share of between 90% and 100% (>90%), exploits were available in the last two years.

Interestingly, exploit availability does not discriminate against less prevalent or rare products. For example, 22% of the 46 products with a market share of between 10% and 20% (>10%) had exploit material available. It also is worth highlighting that the 3% of the products with exploits in the 0-10% market share category actually belong to the category containing more than 9,000 products.

Furthermore, it should be noted that the lowest market share of the products in the Top-50 software portfolio is a considerable 32%. Significantly, the findings also reveal that 26 of the Top-50 programs contained vulnerabilities in the last two years and for 21 (or 80%) of those, exploit material was also available. A complete list of the Top-50 products along with vulnerability and exploit availability can be found in Appendix 5 of this report.

This data turns the common assumption that uncommon products are either unexposed or less exposed to attack, on its head. Thus, identifying vulnerable programs and patching them, regardless of prevalence, is a security-critical priority on both a corporate and private level. By only patching a few common or preselected programs, infrastructures, networks, and end-points can be left highly exposed to attack.

This data turns the common assumption that uncommon products are either unexposed or less exposed to attack, on its head.

Moving Targets -

Knowing What To Patch

4

Dynamic threats require agile, strategic tactics

A strategy of patching a limited and statically defined set of programs considered business-critical fails to reduce the true risks due to the rapidly changing threat environment. For example, what one organisation considers a business-critical program may not be of interest to cybercriminals, who may choose a less popular program to target instead.

Many organisations follow the strategy of defining, in a given year, a static set of programs deemed critical, and thereafter focus their limited resources on only patching these programs to attain the desired risk level. However, the assumptions of such a strategy quickly become out of date due to the fast changing threats of the IT security ecosystem. Many programs found to be critical in a given year are not vulnerable the next year, and vice versa. Therefore, any strategy of patching a limited and static set of programs will undoubtedly suffer from the following drawbacks:

A: **Missing critical programs**
A considerable number of programs not considered critical by the static list will become critical in a subsequent year due to new vulnerabilities. Not covering these programs exposes entire organisations to the risk of compromise – and invalidates the risk assumptions of their strategy.

B: **Patching non-critical programs**
Deploying patches (feature updates) or patching programs with low criticality ratings while other programs with "Highly critical" vulnerabilities remain unpatched, is a waste of valuable security resources as the risk is not reduced as expected.

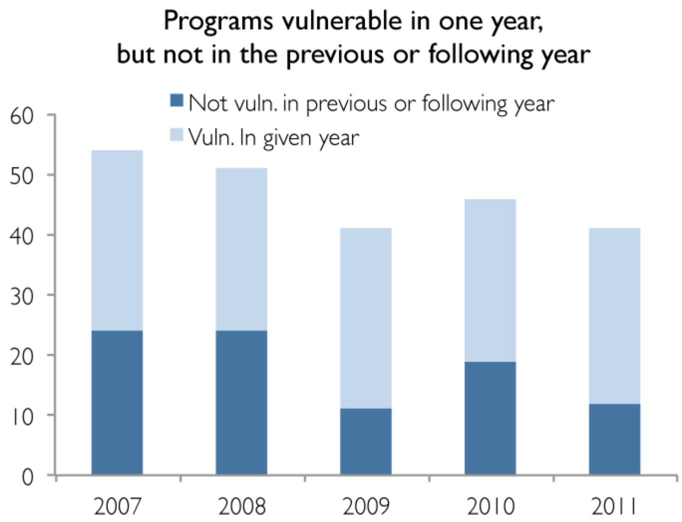
To what degree programs deemed critical in a given year turned out not to be critical the following year, and vice versa, was therefore analysed and quantified. The optimal strategy to achieve the highest reduction of risk given limited security resources was then formulated.

For the purpose of this analysis, a software portfolio that is representative of a small organisation and comprises the Top-200 most prevalent programs found by the Secunia PSI was reviewed. The least prevalent program in this portfolio has a market share of 5.8%. To test the validity of the previously stated static strategy, the number of programs that contained vulnerabilities in a given year, but not in the previous or following year, was analysed.

Table 4 and the left pane of Figure 12 show the results for the Top-200 software portfolio for the period from 2007 to 2011.

Top-200 Portfolio Programs…	2007	2008	2009	2010	2011	Average
... vulnerable in a given year	54	51	41	46	41	**47**
... not vulnerable in the previous ... or following year	24 (44%)	24 (47%)	11 (27%)	19 (41%)	12 (29%)	**18** **(39%)**

Table 4 – Number of programs of the Top-200 portfolio found vulnerable in a given year, and the number found not to be vulnerable in the previous or following year.



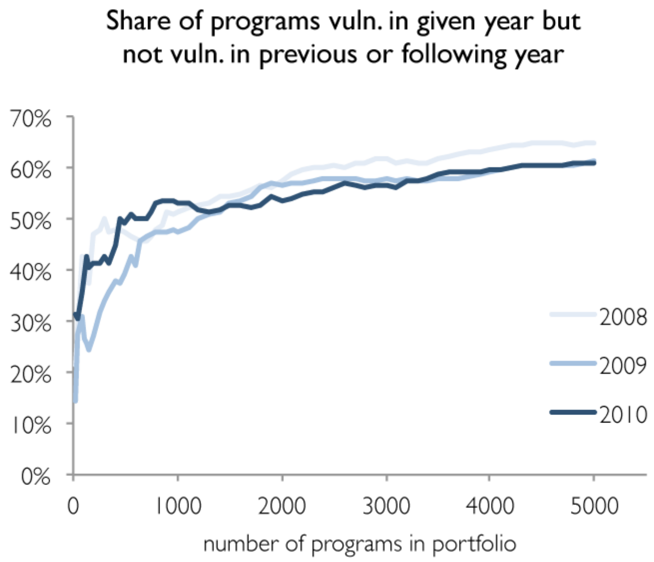


Figure 12 – Share of programs with vulnerabilities in a given year and the share of these that are not vulnerable in the previous or following year.

On average, 47 of the 200 programs in the sample portfolio had vulnerabilities in any given year. For example 46 of the 200 programs had vulnerabilities in 2010. As it is impossible to foresee how many programs will be vulnerable in 2012, the numbers for 2011 are less accurate; however the information for the previous five years is complete.

Of these vulnerable programs, on average 18 or 39% were not vulnerable in the previous or the following year. For example, for the 46 programs found to be vulnerable in 2010, a considerable 41% (19 programs) were not vulnerable in the previous year or the following year. Thus, a strategy that identified the 46 vulnerable programs in 2010 as critical is already considerably out-of-date in the following year.
The results depicted in Figure 12 illustrate that identifying the critical programs worth patching is a dynamic process, similar to chasing a continually moving target. While some programs are vulnerable in several consecutive years, many programs are only vulnerable in certain years.

As previously stated; even programs with low prevalence are also frequently affected by vulnerabilities and have exploit material available. Some programs (the usual suspects such as the operating system, web browsers, and a few others) contain vulnerabilities in every year. However, as the data shows, there are a considerable number of programs that are vulnerable in one year and not in another, and vice versa. This observation is found to be valid for each of the previous five years.

Therefore, despite the fact that on average only 47 of the 200 programs in the sample portfolio had vulnerabilities in the last five years, the monitoring of all 200 programs is required in order to correctly assess the risk and deploy patches that remediate the critical risks. Any static approach of defining critical programs is doomed to fail by becoming rapidly outdated due to the unpredictability and highly dynamic nature of the threats. Efficient identification of the truly vulnerable programs, and patching those first, is the right approach to achieve the largest risk reduction given limited resources

The risks for enterprise portfolios

Medium to large organisations typically have hundreds or thousands of unique programs installed in their infrastructures. To assess the impact of the portfolio size (number of unique programs deployed in a given organisation), the percentage of the programs identified as vulnerable in a given year but not vulnerable in the previous or following year for portfolio sizes from 25 to 5,000 programs, was analysed.

Figure 12 (right pane) shows that the percentage increases with the increasing portfolio size. For example, for a portfolio with 1,000 programs, 93 programs were found to be vulnerable in 2010. However, 53% of these same programs were not vulnerable in the subsequent years. This percentage increased to 61% for a portfolio with 5,000 programs.

Thus, for organisations with more than 600 programs in their portfolio more than half of the programs that are vulnerable in one year are not vulnerable in the previous or the following year.

To assess and effectively remediate these risks it is therefore imperative to monitor all 600 programs in the portfolio.

The larger the organisation, the more important it becomes to dynamically identify vulnerable programs in order to remediate the most critical risks – deploying the patches that result in the largest reduction of risk. This approach becomes more important under the assumption of limited security resources.

To master these challenges it is imperative for organisations to deploy tools and have clearly defined processes to A) identify all programs in their infrastructure, B) correlate this information with up-to-date data about vulnerabilities, and C) have the processes in place to deploy critical patches in a timely fashion.

Thus, for organisations with more than 600 programs in their portfolio more than half of the programs that are vulnerable in one year are not vulnerable in the previous or the following year.

Appendix & Glossary

1. Secunia Vulnerability Tracking Process

A vulnerability is an error in software which can be exploited with a security impact and gain. Secunia validates, verifies, and tests vulnerability information gathered and includes it in the Secunia Vulnerability Intelligence database with consistent and standard processes, which have been constantly refined over the years.

Whenever a new vulnerability is reported, a Secunia Advisory is released after verification of the information. A Secunia Advisory provides details, including description, risk rating, impact, attack vector, recommended mitigation, credits, references, and more for the vulnerability including additional details discovered during verification and testing, thus providing the information required to make appropriate decisions about how to protect systems. After the first publication, the status of the vulnerability is tracked throughout its lifecycle and updates are made to the corresponding Secunia Advisory as new relevant information becomes available.

2. Secunia Vulnerability Criticality Classification

The criticality of a vulnerability is based on the assessment of the vulnerability's potential impact on a system, the attack vector, mitigating factors, and if an exploit exists for the vulnerability and is being actively exploited prior to the release of a patch.

Extremely Critical (5 of 5)	Typically used for remotely exploitable vulnerabilities that can lead to system compromise. Successful exploitation does not normally require any interaction and exploits are in the wild. These vulnerabilities can exist in services like FTP, HTTP, and SMTP or in certain client systems like email programs or browsers.
Highly Critical (4 of 5)	Typically used for remotely exploitable vulnerabilities that can lead to system compromise. Successful exploitation does not normally require any interaction but there are no known exploits available at the time of disclosure. Such vulnerabilities can exist in services like FTP, HTTP, and SMTP or in client systems like email programs or browsers.
Moderately Critical (3 of 5)	This rating is also used for vulnerabilities allowing system compromise on LANs in services like SMB, RPC, NFS, LPD and similar services that are not intended for use over the Internet. Typically used for remotely exploitable Denial of Service vulnerabilities against services like FTP, HTTP, and SMTP, and for vulnerabilities that allow system compromises but require user interaction.
Less Critical (2 of 5)	Typically used for cross-site scripting vulnerabilities and privilege escalation vulnerabilities. This rating is also used for vulnerabilities allowing exposure of sensitive data to local users.
Not Critical (1 of 5)	Typically used for very limited privilege escalation vulnerabilities and locally exploitable Denial of Service vulnerabilities. This rating is also used for non-sensitive system information disclosure vulnerabilities (e.g. remote disclosure of installation path of applications).

3. Attack Vector

The attack vector describes the way an attacker can trigger or reach the vulnerability in a product. Secunia classifies the attack vector as "Local system", "From local network", or "From remote".

Local System	Local system describes vulnerabilities where the attacker is required to be a local user on the system to trigger the vulnerability.
From Local Network	From local network describes vulnerabilities where the attacker is required to be situated on the same network as a vulnerable system (not necessarily a LAN). This category covers vulnerabilities in certain services (e.g. DHCP, RPC, administrative services) that should not be accessible from the Internet, but only from a local network or optionally from a restricted set of external systems.
From Remote	From remote describes other vulnerabilities where the attacker is not required to have access to the system or a local network in order to exploit the vulnerability. This category covers services that are acceptable to be exposed and reachable to the Internet (e.g. HTTP, HTTPS, SMTP). It also covers client applications used on the Internet and certain vulnerabilities where it is reasonable to assume that a security conscious user can be tricked into performing certain actions.

4. Genuine and Shared Vulnerabilities

Genuine Vulnerabilities	Vulnerabilities found in the software of this and only this vendor. These are vulnerabilities in the code developed by this vendor that are not shared in the products of other vendors.
Shared Vulnerabilities	Vulnerabilities found in the software of this and other vendors due to the sharing of either code, software libraries, or product binaries. If vendor A develops code or products that are also used by vendor B, the vulnerabilities found in these components are genuine for vendor A and counted as shared vulnerabilities for vendor B.
Total Vulnerabilities	The total number of vulnerabilities found in the products of the vendor, be it genuine or shared vulnerabilities. These are the vulnerabilities that affect the users of the vendor's products.

5. The Top-50 Software Portfolio

The following table lists the programs in the Top-50 software portfolio together with the type of program (MS Microsoft, TP third-party), market share as of December 2011, the number of vulnerabilities (CVEs) affecting the program in 2010 and 2011, and whether exploit material was made available for the program in this period.

The ranking and market share is derived from anonymous scans of the Secunia PSI[12] in December 2011. Note that the sum of the vulnerabilities in this table does not reflect the total number of vulnerabilities in the portfolio as many products share vulnerabilities.

For example Adobe Flash Player (#4), Adobe Reader (#8), and Adobe AIR (#21) share code components and thereby also share numerous vulnerabilities. For each program the unique number of CVEs of this given program in the given year is listed.

Exploit availability indicates that at least one exploit for the specified program was available.

Programs 1-18 of the Top-50 software portfolio

Rank	Type	Program	Share	CVEs 2010	CVEs 2011	Exploit Avbl.
1	ms	Microsoft XML Core Services (MSXML)	100%	1	0	Yes
2	ms	Microsoft Internet Explorer	99%	54	38	Yes
3	ms	Microsoft .NET Framework	99%	5	10	Yes
4	tp	Adobe Flash Player	98%	56	63	Yes
5	ms	Microsoft Visual C++ Redistributable	94%	1	0	
6	ms	MSCOMCTL ActiveX Control	86%	0	0	
7	tp	Sun Java JRE	85%	58	58	Yes
8	tp	Adobe Reader	82%	116	117	Yes
9	ms	Microsoft Silverlight	78%	2	2	
10	ms	Microsoft Windows Defender	73%	0	1	
11	ms	Microsoft Word	72%	16	0	Yes
12	ms	Microsoft Excel	70%	36	24	Yes
13	ms	Microsoft PowerPoint	68%	10	7	Yes
14	ms	Windows DVD Maker	66%	0	0	
15	tp	Mozilla Firefox	63%	95	96	Yes
16	ms	Microsoft Malicious Software Removal Tool	62%	0	1	
17	tp	Apple Software Update	55%	0	0	
18	tp	comdlg32 ActiveX Control	54%	0	0	

12: The Secunia PSI is a free, lightweight scanner that identifies and patches insecure programs on end-points - http://secunia.com/PSI

Programs 19-50 of the Top-50 software portfolio

Rank	Type	Program	Share	CVEs 2010	CVEs 2011	Exploit Avbl.
19	ms	Microsoft Outlook	53%	2	0	Yes
20	tp	Apple QuickTime	52%	34	29	Yes
21	tp	Adobe AIR	52%	38	28	Yes
22	ms	Driver Package Installer (DPInst)	50%	0	0	
23	tp	Microsoft Office (extension for Firefox)	49%	0	0	
24	ms	Microsoft Windows Media Player	48%	2	0	Yes
25	tp	Java Console 6.x (extension for Firefox)	48%	0	0	
26	tp	CCleaner	48%	0	0	
27	ms	Microsoft PowerPoint Viewer	47%	2	4	Yes
28	tp	Google Chrome	45%	147	321	Yes
29	ms	Windows Live Messenger	44%	0	0	
30	ms	Microsoft Access	44%	2	0	Yes
31	ms	Windows Live	42%	1	0	Yes
32	ms	Microsoft Publisher	42%	7	4	Yes
33	tp	Realtek Voice Manager	41%	0	0	
34	ms	CAPICOM	41%	0	0	
35	tp	Adobe Updater	38%	0	0	
36	tp	Skype	37%	0	2	
37	tp	Google Earth	37%	0	0	
38	tp	Apple iTunes	37%	63	138	Yes
39	tp	VLC media player	37%	7	10	Yes
40	ms	Microsoft Windows Genuine Advantage ActiveX Control	36%	0	0	
41	tp	ITDetector ActiveX Control	36%	0	0	
42	ms	Windows Live Essentials	36%	0	0	
43	ms	Microsoft Office Picture Manager	36%	0	0	
44	ms	Microsoft Office Template and Media Control ActiveX Control	36%	0	0	
45	tp	Realtek AC 97 Update and remove driver Tool	35%	0	0	
46	tp	InstallShield Update Service	35%	0	0	
47	ms	Microsoft Visio Viewer	34%	0	0	
48	tp	Apple Bonjour for Windows	33%	0	0	
49	ms	Windows Live Photo Gallery 2011	32%	0	0	
50	ms	Windows Live Movie Maker 2011	32%	0	0	

Glossary

Vulnerability
A vulnerability is an error in software which can be exploited with a security impact and gain.

Exploit
Malicious code that takes advantage of software vulnerabilities to infect a computer or perform other harmful actions.

Botnet
A botnet is a collection of compromised computers connected to the Internet (each compromised computer is known as a 'bot'). When a computer is compromised by an attacker, there is often code within the malware that commands it to become part of a botnet. The "botmaster" or "bot herder" controls these compromised computers. Computers in a botnet are often called nodes or zombies.

0-Day Exploit (aka Zero Day Exploit)
A 0-day is a vulnerability which is being actively exploited, prior to the public release of details about the vulnerability.

Malware
Any software that is designed specifically to cause damage to a user's computer, server, or network. Viruses, worms, and Trojans are all types of malware.

Social Engineering
A technique that defeats security precautions by exploiting human vulnerabilities. Social engineering scams can be both online (such as receiving email messages that ask the recipient to click the attachment, which is actually malware) and offline (such as receiving a phone call from someone posing as a representative from one's credit card company). Regardless of the method selected, the purpose of a social engineering attack remains the same - to get the targeted user to perform an action of the attacker's choice.

Trojan
A generally self-contained program that does not self-replicate but takes malicious action on the computer.

Secunia
Stay Secure

For further information, please visit
secunia.com

Secunia
Mikado House
Rued Langgaards Vej 8
DK-2300 Copenhagen S
Denmark

Email: info@secunia.com
Phone: +45 7020 5144
Fax: +45 7020 5145

Copyright 2012 Secunia. All rights reserved.
This report may only be redistributed unedited and unaltered.
This report may be cited and referenced only if clearly crediting Secunia and this report as the source. Any other reproduction and redistribution in print or electronically is strictly prohibited without explicit permission.