IBM X-Force® 2010
Mid-Year Trend and Risk Report

August 2010

IBM

Contributors

Producing the X-Force Mid-year Trend and Risk Report is a dedication in collaboration across all of IBM. We would like to thank the following individuals for their rapt attention and dedication to the publication of this report.

Contributor	Title
Bryan Williams	X-Force Research and Development, Protection Technologies
Carsten Hagemann	X-Force Software Engineer, Content Security
Dr. Jens Thamm	Database Management Content Security
Frank (Jamie) Licitra	X-Force Product Manager
Harold Moss	Security Strategy - Emerging Tech & Cloud Computing Technical Architect
Jon Larimer	X-Force Advanced Research, Malware
Leslie Horacek	X-Force Threat Response Manager
Marc Noske	Database Administration, Content Security
Mark E. Wallis	Senior Information Developer, X-Force Database Team
Michael Waidner	CTO for Security, IBM Security Strategy
Michelle Alvarez	Team Lead, MSS Intelligence Center (aka Eagle Eyes)
Mike Warfield	Senior Wizard, X-Force
Ralf Iffert	Manager, X-Force Content Security
Ravi Srinivasan	IBM Software Group, Tivoli Senior Product Manager
Robert Freeman	Senior Technologist & Web Exploit Watchman
Ryan McNulty	IBM Managed Security Services & SQL Querier Extraordinaire
Scott Moore	X-Force Software Developer & X-Force Database Team Lead
Tom Cross	Manager, X-Force Advanced Research
Wangui McKelvey	X-Force Product Marketing Manager

About X-Force

The IBM X-Force® research and development teams study and monitor the latest threat trends including vulnerabilities, exploits and active attacks, viruses and other malware, spam, phishing, and malicious Web content. In addition to advising customers and the general public on how to respond to emerging and critical threats, X-Force also delivers security content to protect IBM customers from these threats.

Contents
Section I

Contents
Section II

Overview

As we move past the mid-point of 2010 and into the second half of the year, one thing in this vastly changing world remains constant: attackers continue to take advantage of the rapid pace of technology for financial gain, including theft of intellectual property. At the end of 2009, we summarized the evolution of the threat landscape for both security professionals and attackers. More technology, better automation, and a more manageable user experience sums up the tools that each side uses. We saw the rise of designer malware with rich feature sets that match the sophistication of commercial software. Rather than focusing on a single point of entry, these latest threats aggressively target multiple resources within an enterprise to ensure successful exploitation. No longer are single, public-facing resources the greatest risk, but instead, every employee and endpoint has become a potential point of entry. Sophisticated combinations of vulnerability exploitation, spam, phishing, malicious URLs and social engineering are all easier to obfuscate, automate, and deploy than ever before.

Enterprises and the global economy have been in transition. Companies merge divisions and scale down the size of their organizations as new technologies help simplify tasks. Through all these shifts and changes within the micro-climate of organizations, we understand the confusion it can cause and the adaptation demands that it places on the work force. What must we protect? As we grow into new markets and adopt new technology, how has the security outlook changed?

We have seen traditional security solutions become wholly ineffective against new methods of obfuscation and low-volume attack vectors. Attacks targeting Web servers via SQL injection and cross-site scripting are nothing new, but they continue to be creatively concealed to bypass many security products. Employees are directly targeted through the documents they work with every day—whether as PDF files or office documents.

Threat dynamics continue to evolve at a furious pace making it even more crucial to look at unfolding trends so we can better prepare ourselves for the future.

New Layout and design

We have redesigned the structure and layout of this year's mid-year report to contain two main sections. The first section covers hot topics and the newest major trends while the second section covers our more traditional content—in-depth threat data along with the thoughtful analysis that our readers have come to expect from IBM Security Solutions.

2010 Mid-year highlights

Vulnerabilities and exploitation

- Advanced persistent threat—What concerns X-Force most about these sophisticated attackers is their ability to successfully penetrate well-defended networks in spite of significant advances in network security technology and practices. In particular, we are concerned about increasingly obfuscated exploits and covert malware command-and-control channels that fly under the radar of modern security systems.
- Obfuscation, obfuscation, obfuscation—Attackers continue to find new ways to disguise their malicious traffic via JavaScript and PDF obfuscation. Obfuscation is a technique used by software developers and attackers alike to hide or mask the code used to develop their applications. Things would be easier if network security products could simply block any JavaScript that was obfuscated, but unfortunately, obfuscation techniques are used by many legitimate websites in an attempt to prevent unsophisticated Web developers from stealing their code. These legitimate websites act as cover for the malicious ones, turning the attacks into needles in a haystack.
- PDF attacks continue to increase as attackers trick users in new ways. To understand why PDFs are targeted, consider that endpoints are typically the weakest link in an enterprise organization. Attackers understand this fact well. For example, although sensitive data may not be present on a particular endpoint, that endpoint may have access to others that do. Or, that endpoint can be used as a practical bounce point to launch attacks on other computers.
- Reported vulnerabilities are at an all time high—2010 has seen a significant increase in the volume of security vulnerability disclosures, due both to significant increases in public exploit releases and to positive efforts by several large software companies to identify and mitigate security vulnerabilities.
- Web application vulnerabilities have inched up to the 55 percent mark, accounting for fully half of all vulnerability disclosures in the first part of 2010.
- Exploit Effort versus Potential Reward—What are attackers really going after? With the number of vulnerability announcements rising and vendors scrambling to provide patches and protection to problem areas, how can enterprises best prioritize the efforts of IT administrators to provide adequate coverage? The Exploit Effort versus Potential Reward Matrix provides a simple model for thinking about vulnerability triage from the perspective of attackers.

Malware and the malicious Web

- The Conficker worm was one of the biggest computer security stories of the past few years, so an update for this trend report is clearly in order. What has happened to the Conficker worm since 2009?
- The Zeus botnet toolkit continues to wreak havoc on organizations. Early 2010 saw the release of an updated version of the Zeus botnet kit, dubbed Zeus 2.0. Major new features included in this version provide updated functionality to attackers.
- BlackHat SEO and Rogue antivirus exploits still penetrate enterprises by tricking end users.
- Malicious Web toolkits—The continued prevalence of the Gumblar toolkit is helping to secure a top exploit position for Adobe products, but PDF and Flash exploits are also extremely popular in many other exploit toolkits. An interesting change from the second half of 2009 is that ActiveX has dropped off the top-five list, at least for now.

Spam and phishing

- The top spam domains have moved from China (**.cn**) to Russia (**.ru**).
- Since mid-March of 2010, the average size of spam doubled without any changes in the percentage of image-based spam. In the following weeks, the average spam byte size continued to increase until the beginning of June, reaching an average size of nearly 10 KB.
- In the first half of 2010, financial institutions remain the number one target, but now represent only about 49 percent of all phishing email targets.
- More than two thirds of all financial phishing targets in the first six months of 2010 are located in North America; the remaining 32 percent are located in Europe.
- Brazil remains the top sender in terms of phishing volume, while India is in second place, and South Korea holds third place.

Future topics beyond 2010

- Virtualization—Organizations are under increasing pressure to deliver more functionality to their businesses and customers. At the heart of this transformation is virtualization. However, the ultimate success of virtualization depends not only on energy efficiency, performance, and ease of use, but also on the ability to provide these benefits without compromising the overall security, reliability, and availability of the IT infrastructure.
- IPv6 deployments—What is accelerating the trend to adopt these new networks?
- Cloud computing is an emerging technology in which the vulnerabilities today are identical to those found in traditional emerging technology, compounded by the challenges of everyday remote management activities. Cloud computing is in its relative infancy and is multi-faceted as it relates to implementation and scope based on design and utilization.

IBM Security collaboration

IBM Security represents several brands that provide a broad spectrum of security competency. While the X-Force® research and development teams are busy at work analyzing the latest trends and methods used by attackers, other groups within IBM work to supply that rich data into protection techniques for our customers.

- The IBM X-Force research and development team discovers, analyzes, monitors, and records a broad range of computer security threats and vulnerabilities.
- IBM Managed Security Services (MSS) is responsible for monitoring exploits related to endpoints, servers (including Web servers), and general network infrastructure. MSS tracks exploits delivered over the Web as well as other vectors such as email and instant messaging.
- Professional Security Services (PSS) delivers comprehensive, enterprise-wide security assessment, design, and deployment services to help build effective information security solutions.
- Our “Whiro” crawlers combine alert data from MSS, our Content Security team, and independent analysis to monitor exploitation from Web-based sources. Whiro uses specialized technology to identify exploits used even in the most obfuscated cases, including those cases where toolkits attempt multiple exploits.
- Our Content security team independently scours and categorizes the Web through crawling, independent discoveries, and through the feeds provided by MSS and Whiro.
- IBM has collated real-world vulnerability data from security tests conducted over the past three years from the IBM Rational AppScan onDemand Premium service. This service combines application security assessment results obtained from IBM Rational AppScan with manual security testing and verification.
- IBM Cloud Security Services allows clients to consume security software features through a hosted subscription model that helps reduce costs, improve service delivery, and improve security.
- Identity and access management solutions provide thorough identity management, access management, and user compliance auditing. These solutions centralize and automate the management of users, authentication, access, audit policy, and the provisioning of user services.

Hot trends to understand in 2010

Covert threats to the enterprise

The practice of computer security has been marked in first half of 2010 by the presence of a new term in nearly every conversation: Advanced Persistent Threat. Only recently did the dialog about the nature of the threat to our networks shift from bored groups of teenage computer hackers out for a joyride on the Internet toward professional groups of computer criminals who are in it for the money. Now, it appears that there is an even more insidious threat on the horizon—well funded, state sponsored intelligence organizations. Advanced Persistent Threat is not new—these kinds of attacks have been going on for years. What is new is the wide variety of different kinds of organizations who are talking about this threat and fighting it on their networks.

What concerns X-Force most about these sophisticated attackers is their ability to successfully penetrate well-defended networks in spite of significant advances in network security technology and practices. We are particularly concerned about increasingly obfuscated exploits and covert malware command-and-control channels that fly under the radar of modern security systems. Combating these threats requires the development of new processes and ultimately the adoption of entirely new network security technologies.

Advanced persistent threat

The term Advanced Persistent Threat (APT) originated in U.S. Government circles. It refers to a variety of different groups from different nation states that attack computer networks in order to steal intelligence information, as opposed to groups with a more direct financial motivation, such as those who target caches of credit card numbers. The word persistent is used to characterize the capacity that APT groups have for maintaining access to and control of computer networks even when the network operators are aware of their presence and are taking active steps to combat them. APT groups are patient—they slowly develop access to the information they want while staying below an activity threshold that would attract attention.

The level of sophistication of attack techniques seen in APT cases is often directly proportional to the level of sophistication of the capabilities of the people defending a particular network. APT groups appear to have a library of different tools and capabilities from which they select the least sophisticated capability required to get a particular job done. More sophisticated tools and techniques appear as network defenders discover and react to intrusions.

What all sophisticated, targeted attacks have in common is that the first step for the attacker is reconnaissance. Although this may include the traditional network probing and scanning activities that we associate with computer intrusions, sophisticated attackers think outside of that box.

Today there is a wealth of information available on the Internet regarding many people working in the business world. We publish profiles on personal and professional social networking sites, we send out status updates that indicate where we are traveling, we engage in online forums relevant to our jobs, we talk at public conferences, we write articles and papers, we take news media interviews, and in doing all of these things we leave a large number of bread crumbs that malicious persons can use to reconstruct not just a picture of our own personal lives, but of the organizations that we work for and how we fit into them.

Sophisticated attackers

Sophisticated attackers use this public information to develop a complete picture of a targeted organization; who works there, what they do, and who they report to within the organization. This picture enables them to identify the particular individuals who may have access to the kind of information that they seek. Those individuals are targeted with various kinds of social engineering attacks intended to trick them into running a malicious exploit. The attacker's initial goal is to gain control of the victim's workstation. From that point, all of the victim's work and communications become an open book.

These attacks often involve malformed documents or Web pages that target zero-day vulnerabilities with obfuscated exploits. The attack might come as an email, addressed from a business partner or colleague, with a malicious attachment that sounds directly relevant to the victim's job function. It might be a link to a juicy document that is hosted on a competitor's website, or perhaps a USB token handed to the victim at a trade show with an interesting presentation.

The custom malware that is installed by the exploit uses covert channels to communicate over the network without being noticed. Once the attackers have their malware running on one victim's machine, they often try to spread their control to other systems in the targeted network. They will also try to exploit business relationships in order to leverage their control over one company's network to break into others.

For network security professionals in the private sector, the line between intelligence-related APT activity and financially motivated attacks is blurry at best. Power plants have been attacked by state-sponsored cyber warriors as well as criminal groups who are simply interested in blackmail. The same sort of sophisticated spear phishing attacks that have been used to target government strategists have also been directed at executives in financial institutions who have access to funds transfer systems.

In some respects this makes our jobs easier—the techniques that we develop to combat these kinds of attacks can apply to a wide variety of contexts. However, it's important to recognize that the term APT does not encompass the whole spectrum of sophisticated, targeted attacks that enterprises are facing. While all of the recent discussion of APT has helped raise awareness about these kinds of attack techniques, we hope that it does not drive a reaction that is too narrowly focused on intelligence-related activities. It is the responsibility of network security practitioners to find ways to protect their networks against these kinds of attacks regardless of what the attacker's motivation might be.

Financially motivated attacks

In the 2008 Annual X-Force Trend and Risk Report we introduced a simple model for thinking about vulnerability triage from the perspective of financially motivated attackers, from which we produced the Exploit Effort versus Potential Reward Matrix (formerly, Exploitability Probability Matrix). This chart plots different security vulnerabilities in terms of the opportunity they represent to computer criminals as well as the effort associated with exploiting them. The chart, which is updated on page 21 of this report, helps illustrate the fact that vulnerabilities achieving widespread exploitation on the Internet tend to fit into a sweet spot—easy to exploit and a big opportunity for the bad guys. These kinds of vulnerabilities are often favored by organized criminal groups who are involved in mass exploitation of large numbers of endpoint systems.

However, some of the vulnerabilities that X-Force publishes alerts and advisories about are relatively expensive to exploit. Sophisticated attackers who have the ability to develop custom attack tools may take advantage of these kinds of vulnerabilities in spite of the cost. Also, particular vulnerabilities tend to become less expensive to exploit over time. In many cases, vulnerabilities are first discovered by sophisticated attackers and used in targeted attacks. Eventually the attack activity is uncovered by security professionals, and the vulnerability is publicly disclosed and patched. As more information about the vulnerability emerges publicly, ultimately including exploit code, the pattern of attack activity associated with that vulnerability moves from targeted attacks to widespread exploitation.

In this regard there is a direct relationship between targeted APT style attacks and widespread botnet activity, in that vulnerabilities and obfuscation techniques that are developed by sophisticated teams for use in targeted attacks eventually trickle down into the mass exploitation toolkits employed by organized criminal groups. The result is that attackers at all levels of the food chain are becoming more sophisticated over time. What is particularly problematic about this evolution is the growing capability that attackers at all levels have to evade the protection offered by various commercial, off-the-shelf network security solutions and to operate under the radar of network managers. These developments put increased pressure on the security industry to become more effective at detecting threats in the real world and not just in the laboratory.

JavaScript obfuscation—a popular evasive technique

The most important example of this sort of evasive technique is JavaScript obfuscation. JavaScript is a flexible language. It allows data to be executed as code, and data can be manipulated. It can be encrypted. In real world attacks, exploit payloads are often delivered via JavaScript, and those exploit payloads are hidden within heavily encoded data portions of the JavaScript, which are too expensive to inspect on the network but are unraveled by browsers and document viewers when they reach the endpoint. Things would be easier if network security products could simply block any JavaScript that was obfuscated, but, unfortunately, obfuscation techniques are used by many legitimate websites in an attempt to prevent unsophisticated Web developers from stealing their code. These legitimate websites act as cover for the malicious ones, turning the attacks into needles in a haystack.

There is no silver bullet that addresses APT. Off-the-shelf security solutions can provide some tools that help, but there is no product you can buy that magically makes this problem go away. Many organizations are evaluating new processes and technologies, such as the wider use of physical network segmentation, universal email signing, and application white-listing. All of these approaches raise the bar, but they cannot make it insurmountable. Fundamental problems such as obfuscation require new technical solutions. The onus is on the security industry to drive innovation in this area to help arm network administrators to respond.

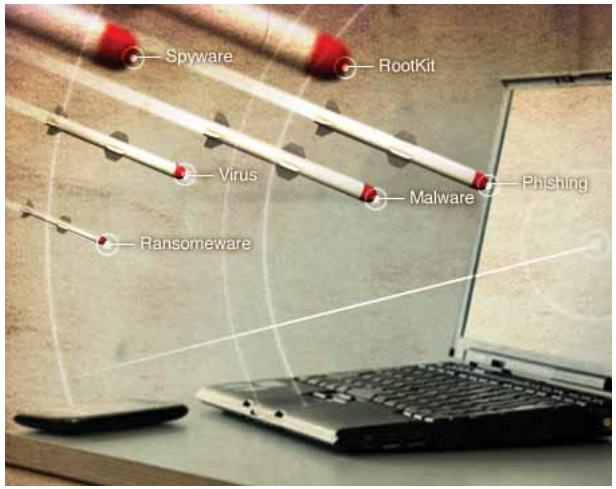
Fighting APT

In our experience, one of the most effective things that you can do to combat this sort of threat on your network is to enlist your people. We reject the idea that it is impossible to train users to be on guard for sophisticated spear phishing attacks, because we've seen it work. If you can identify the people who work in your organization who are most at risk for this kind of attack, and you sit down with them and explain the nature of the threat and how it works, they can become your first line of defense. They can report suspicious emails to you. Once you've got a sample of an exploit being used by these attackers, you've got a foothold on the problem. You may be able to identify other targeted victims, identify malware command and control patterns, and begin to unravel the infestation.

PDF exploitation is HOT!

X-Force started observing widespread use of PDF-based exploits during the first half of 2009. Since then, based on our data, it has captured three of the top five slots for browser exploits used in the wild. To understand why PDFs are targeted, consider that endpoints are typically the weakest link in an enterprise organization, and attackers understand this fact well. For example, even though sensitive data may not be present on a particular endpoint, that endpoint may have access to others and/or can be used as a practical bounce point to launch attacks on other computers. This still does not fully explain why PDFs are so frequently used, especially when Internet Explorer and ActiveX-based attacks had been so prevalent for years.

We can offer some useful speculations here. First, market-share changes for browsers are inconvenient for attackers making specific investments compared to software that is ubiquitous to virtually all browsers, such as Adobe plug-ins like PDF and Flash. Second, the complexity of vulnerabilities involved in browser-specific exploits may be relegating them to targeted attacks. In other words, someone who has invested the time in finding a reliably exploitable IE or Firefox (or other browser) bug is unlikely to sell the vulnerability details and "weaponized" proof-of-concept for the same price as other vulnerabilities that are easier to find.



Are PDF vulnerabilities easier to find? Compared to ActiveX bugs, there is no evidence of this. However, Microsoft has been quite diligent about blacklisting vulnerable ActiveX interfaces, including those from third party vendors, via "kill bits." When considering the complexity of the "dangling pointer" bugs disclosed for IE this year, PDF vulnerabilities have been less involved.

Another advantage to PDF exploitation over browser-specific attacks is that the document specification for PDF is complex, and attackers can easily stuff data away elsewhere in the PDF document to later be retrieved programmatically and put through a decoder algorithm to return malicious script. This obfuscation approach has deviated from earlier techniques that were more or less 1:1 translations from common JavaScript encoding routines used by exploit toolkits. We discuss the evolution of obfuscation over time in our **Web Browser Exploitation Trends** section. While the act of retrieving data stuffed in other objects within the PDF can be suspect at times, advanced technologies to prevent PDF exploitation should not rely too heavily on artifacts for detection as it leads to false positives.

Zero-day PDF attacks enjoy a lead time before patches are available, creating enhanced value for PDF attacks from the attackers' point of view. Adobe is taking an aggressive, proactive role in dealing with attacks these days, and X-Force hopes this trend continues. In addition, there is talk that the next major Acrobat Reader version will contain a sandbox technology to reduce the exploitability of remaining bugs. We shall see how this affects PDF-based attacks in the wild. X-Force believes that it depends on how the technology affects the cost/benefit ratio compared to other browser plug-ins and ubiquitous document and multimedia formats. Alternate PDF viewers are not immune to bugs and—though rarely targeted by attackers—implementation differences may lead to egregious security risks. As an example, the much discussed PDF "launch" feature produced no prompt with the alternate Foxit Reader. Granted, Adobe's implementation could still spoof prompt fields. However, as with Adobe, Foxit is adding increased security features to their product, such as a "safe mode."

Protection against PDF-based attacks

There are some things that users can do to help protect themselves against PDF-based attacks. In Acrobat Reader, it is possible to disable ActionScript (Adobe's extended version of JavaScript) and, while some PDF-based attacks cannot be prevented this way, it is still valuable to disable this feature. Although there are multiple PDF viewers besides the referenced Acrobat Reader, most options or application preferences should expose similar options. Also, it is interesting to consider that other multimedia formats can be embedded in PDF documents, such as videos and Flash movies. There is an option in the Acrobat preferences to disable this feature. X-Force does not think most enterprises require this feature. In most cases, end users won't either.

Looking forward to the second half of 2010 and into 2011, it is difficult to imagine PDF losing traction with attackers. X-Force expects this to be true regardless of the number of PDF vulnerabilities disclosed. For a number of years after release, an ActiveX issue patched in 2006 continued to be used frequently by Web browser attackers. A key unknown is how the upcoming Acrobat sandbox technology might affect known and unknown exploits.

PDF exploitation attack activity

As noted earlier, PDF exploitation is hot. IBM Managed Security Services (MSS) data concurs. We continue to see this exploitation technique dominate the threat landscape. The most significant jump in event activity associated with PDF attacks occurred in April of this year (see Figure 1). Event activity for this month was almost 37 percent higher than the average for the first half of 2010.

This spike can be attributed to the large surge of malicious spam email in circulation during this month. Victims received an email containing a specially-crafted Adobe Acrobat (PDF) file exploiting the /Launch command.

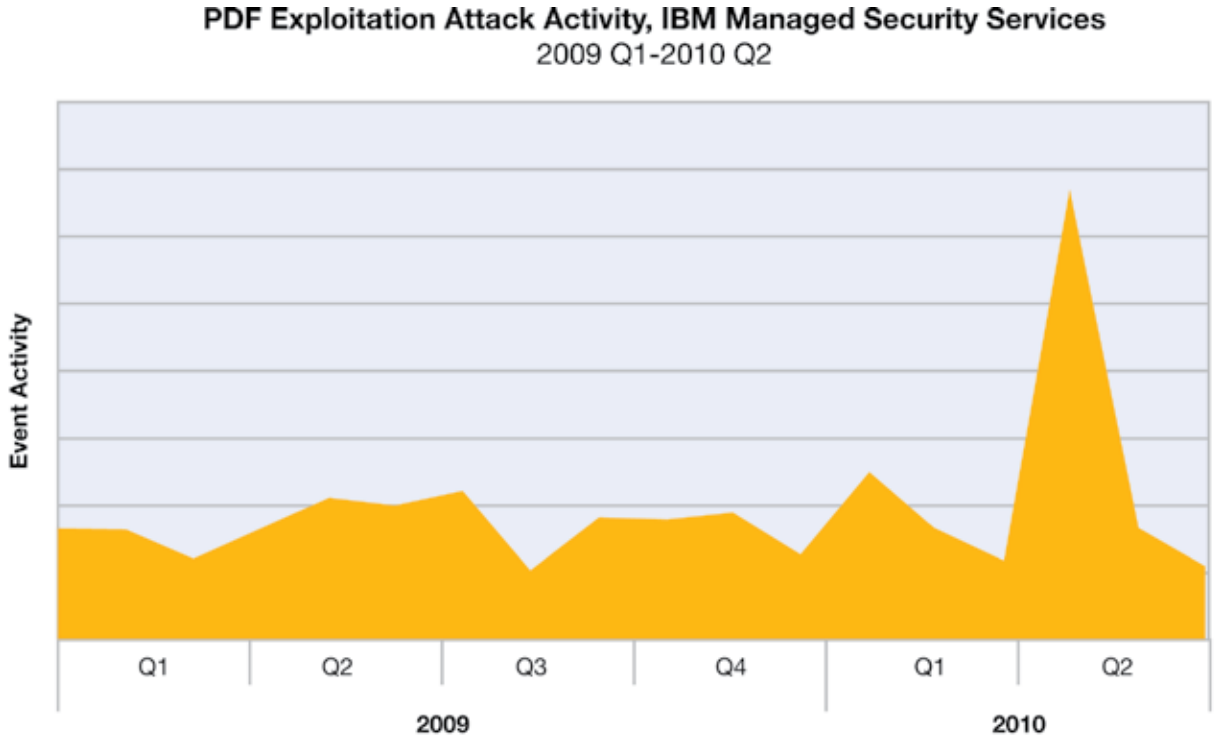


Figure 1: PDF exploitation attack activity, IBM Managed Security Services, 2009 Q1-2010 Q2

IBM Managed Security Services (MSS) provides a view into the most frequently seen types of attacks that leverage client vulnerabilities. MSS offers comprehensive outsourced solutions for real-time security management, including system monitoring, emergency response and 24x7x365 protection.

These services cover a variety of platforms and operating systems for networks, servers, endpoints and wireless applications and provide event monitoring. MSS provides a balanced look at overall attack activity across the Internet. A subset of the MSS data is used in this report to identify attack trends.

The Pushdo, also known as Cutwail, and Zeus botnets had a hand in spreading this PDF malware. Events indicating the network transfer of PDF files containing an embedded action to launch an executable program increased during April 2010. Detections of HTTP messages containing patterns exhibited by the Pushdo Trojan also rose significantly during this month.

And while we are on the topic of Pushdo, IBM X-Force and MSS observed this botnet launching Distributed Denial of Service (DDoS) attacks against certain SSL-enabled websites earlier in the year. Since then, there has been a notable uptick in the detection of specially-crafted messages that could DoS a SSL server. We suspect a majority of this activity can be attributed to Pushdo, which has actually been around since 2007.

We address Zeus botnet facts and myths in depth later in this report.

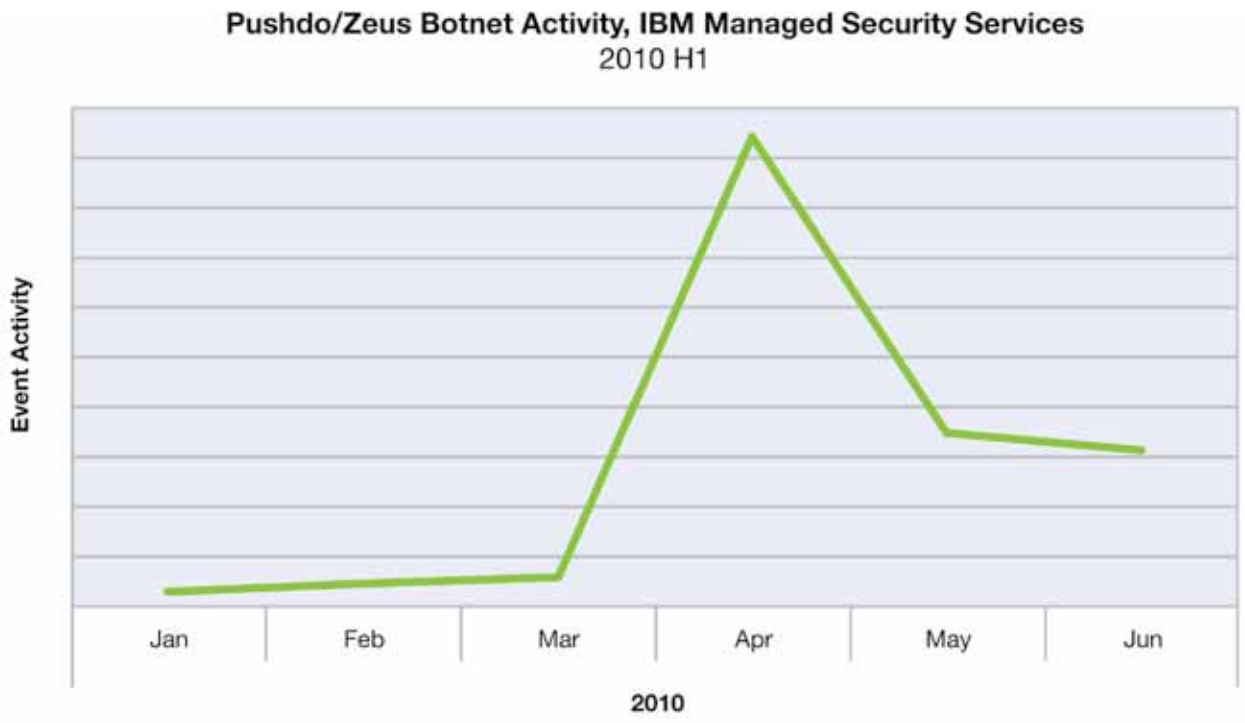


Figure 2: Pushdo/Zeus botnet activity, IBM Managed Security Services, 2010 H1

Malicious code obfuscation trends

The first half of 2010 saw a continuation of the high levels of malicious code obfuscation that existed in 2009. In 2009, most of the JavaScript obfuscations revolved around deriving the malicious script decoding key from the script itself, so any modifications made for analysis yielded gibberish. In terms of JavaScript obfuscation, we have seen some interesting tricks. One trick involves defeating analysis tools with different object scoping by placing a function pointer in an object and retrieving it a different way. Another interesting trick involves a bit of script code that checks the state of an object or image before running further code.

Exploit toolkit packages, discussed in further detail later, continue to favor malicious Adobe Flash and PDF, along with Java files. Obfuscations are developed specifically for these formats. Historically, the obfuscation code is borrowed from earlier JavaScript-based implementations. In 2010, it is becoming increasingly common to use facilities specific to the formats involved in order to hinder analysis. In the case of PDF documents, for example, there are many objects that can contain text and later be accessed programmatically via ActionScript (basically JavaScript). Attackers do not typically package their malicious script in other objects in plaintext, so when they use this obfuscation approach, they almost always use it in conjunction with a decoder algorithm—perhaps ones seen in older toolkits.

Using Visual Basic Script (VBS) as an obfuscation approach continues to decline. Our data indicated a prevalence of 3.6 percent for 2009. For the first half of 2010, we have observed a drop to only 2 percent. We have been discussing VBS use over the last few years as it is a proprietary language only supported by Internet Explorer and thus has historically been a valid obfuscation approach, in large part due to the lack of open-source VBS processing projects. However, while it is not clear why there continues to be a drop in VBS use, this might be a permanent trend.

During the second half of 2009, we observed a potentially emerging trend of using code comments to foul up detection heuristics and to visually obscure the underlying code. When this technique is used, we often see a comment string inside of function call parameters. During the first half of 2010, this technique has not been appearing regularly. X-Force expects this obfuscation approach to be a cyclical fad.

What is obfuscation?

The dictionary meaning of the word "obfuscate" is to make obscure or unclear. To muddy the water if you will.

Within programming language, both software companies and attackers attempt to hide their work. Why do software companies hide or obfuscate code? To protect intellectual capital or protect the logic of the program from reverse engineering or to prevent tampering.

At a high level, this is similar to the way one might use a secret code to prevent others from viewing a private message.

The reason attackers can successfully employ these well known standards to hide their activities, is because many security products cannot interpret every possible encoding/decoding combination and will not detect the attack. This allows for new attack methods that must constantly be reviewed in order to provide detection.

Obfuscated attack activity

The high levels of obfuscation observed in 2009 continue in the first half of 2010. While obfuscated attack activity for the first four months of this year was relatively flat, there was a significant jump in June 2010. Event volume during this month rose to almost 1.4 times the average for H1 2010.

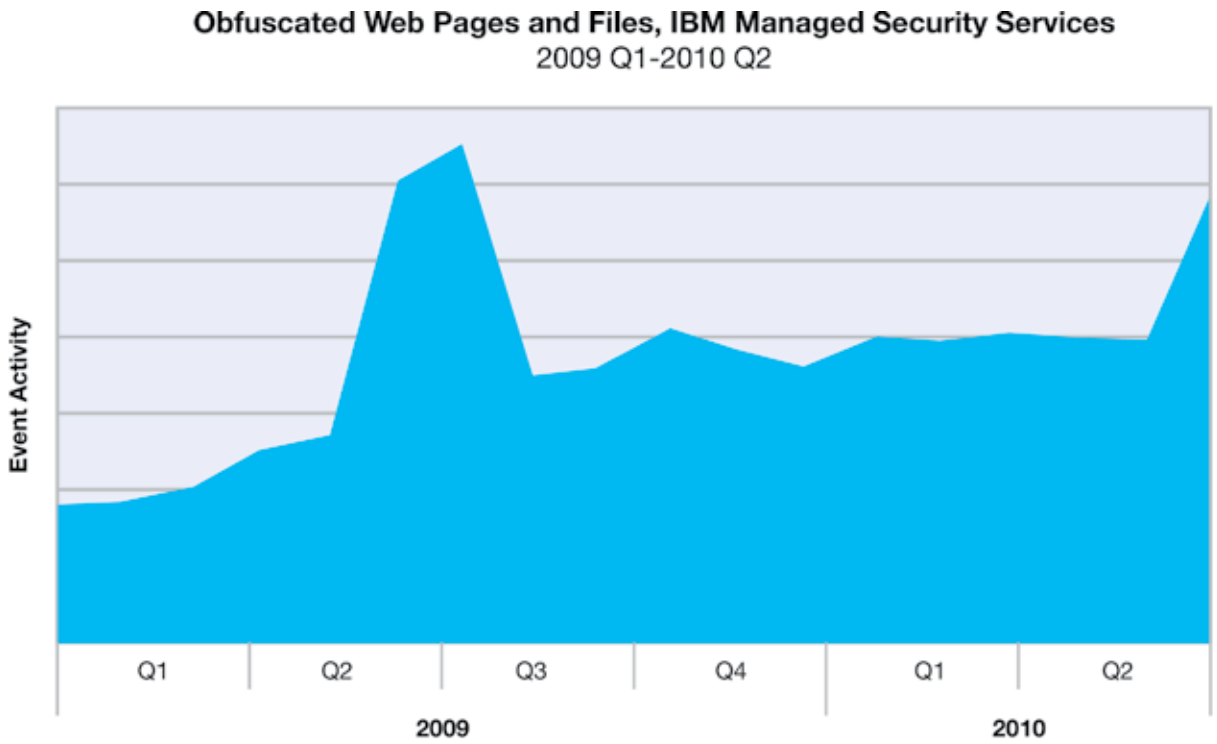


Figure 3: Obfuscated Web pages and files, IBM Managed Security Services, 2009 Q1-2010 Q2

The ever changing threat landscape

Vulnerability disclosures—2010 first half reports well ahead of 2009 numbers

First half of 2010 vulnerability disclosure count

X-Force analyzed and documented 4,396 new vulnerabilities in the first half of 2010, a 36 percent increase compared to the first half of 2009 and the highest count of new disclosures in the first half of the year ever recorded.

In 2007, the vulnerability count dropped for the first time, but in 2008 there was a new record high. While 2009's lower vulnerability disclosures rate appeared to indicate a plateau, the dramatic increase in the first half of this year puts that trend into question. It now looks like 2009 was but a short lull in the ongoing saga of increasing vulnerability disclosures. If the trend from the first half of the year continues, 2010 will bring a new record high.

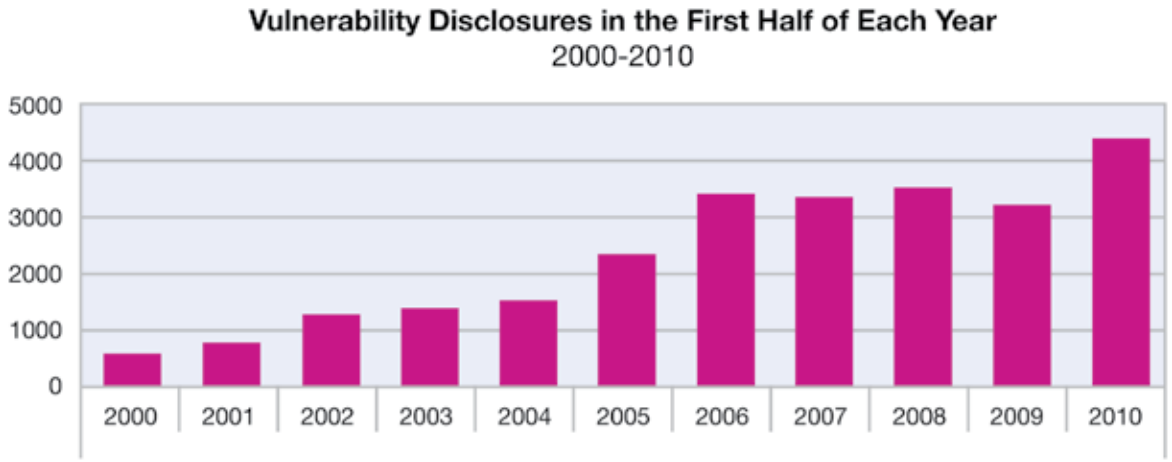


Figure 4: Vulnerability disclosures in the first half of each year, 2000-2010

What does this massive increase in vulnerability disclosures mean? One thing we know for certain—all vendors and other sources are reporting more vulnerabilities than ever before. For example, in 2009 milw0rm disclosed over 2000 exploits. They closed late in that year when the Offensive Security Exploit Database took over. Thus far in 2010, Offensive Security has disclosed over 2000 exploits. That single source alone is trending to release 60 percent more exploits for the year 2010 than in previous years.

The annual vulnerability disclosure rate now appears to be fluctuating between 6,000 and 8,000 new disclosures each year.

To avoid any ambiguity regarding the characterization of vulnerabilities, this report uses the following IBM Security Services definition.

Vulnerability is defined as a set of conditions that leads or may lead to an implicit or explicit failure of the confidentiality, integrity, or availability of an information system.

Patch rate

Over half (55 percent) of all vulnerabilities disclosed in the first half of 2010 have no vendor-supplied patch at the end of the period. This is slightly higher than the 52 percent that applied to all of 2009.

Availability of vulnerability fixes and patches

The top 10 vendors with the most vulnerability disclosures did significantly better than this 55 percent rate, ranging between three and 24 percent of disclosures that were not patched. Table 1 lists the top 10 vendors with the most vulnerability disclosures along with their patch rates for the first half of 2010 and for all of 2009.

X-Force discovered issues with the methodology used in the 2009 end of year report which resulted in flawed findings. In the 2010 mid-year report, we have corrected this methodology so the formulas we are using now are more accurate. We have applied this new methodology to our 2009 data here to gain improved accuracy.

This comparison provides some interesting results. Although Sun had an excellent patch rate of 2.6 percent for 2009, during the first half of 2010 it appears at the top of the unpatched rate at 24 percent. Microsoft comes in at a close second with 23.2 percent of disclosed vulnerabilities unpatched.

Top 10 Vulnerable Vendors and Patch Rates 2010 H1	
Vendor	**% Unpatched**
Sun	24.0%
Microsoft	23.2%
Mozilla	21.3%
Apple	12.9%
IBM	10.3%
Google	8.6%
Linux	8.2%
Oracle	6.8%
Cisco	6.0%
Adobe	2.9%

Top 10 Vulnerable Vendors and Patch Rates 2009	
Vendor	**% Unpatched**
Microsoft	15.8%
HP	14.5%
Mozilla	12.1%
Apple	9.7%
Cisco	8.9%
Linux	5.0%
IBM	4.3%
Oracle	3.3%
Sun	2.6%
Adobe	2.0%

Table 1: Percentage of unpatched vulnerabilities for vendors with the most disclosures in 2010

In general, the percentage of unpatched vulnerabilities in the first half of 2010 is much higher than those rates for the full year of 2009. That may indicate a lower patch rate trend or it may just be that our data cutoff date at the end of June does not reflect the trend for the entire year. Time will tell.

Currently, Adobe is the only vendor in the top ten that has broken into the “less than five percent” category for the first half of 2010 with an impressive rate of only 2.9 percent of disclosed vulnerabilities going unpatched.

Best and worst patchers

Table 2 shows the percentage of disclosures with no patches in the first half of 2010 along with the percent of critical and high disclosures with no patches. Web application platforms (such as WordPress and Joomla!) are excluded from this analysis.

The best and worst patchers chart reflects publicly reported information as catalogued in our database and may not reflect situations where vendors have silently patched vulnerabilities or have assessed public vulnerability reports as inconsequential without issuing a public response to that effect.

This table is sorted by those vendors with the highest percentages of vulnerability disclosures without patches.

Vendor	Percent of 2010 H1 Disclosures with No Patch	Percent of Critical & High 2010 H1 Disclosures with No Patch
All Vendors - 2010 H1 Average	55%	71%
Microsoft	23%	7%
Mozilla	17%	4%
Apple	12%	0%
IBM	9%	29%
Sun	8%	0%
Oracle	7%	22%
Cisco	6%	2%
Novell	5%	10%
HP	4%	5%
Linux	3%	0%
Adobe	3%	2%
Google	0%	0%

Table 2: Best and worst patchers, 2010 H1

Exploit effort versus potential reward matrix

With the number of vulnerability announcements rising and vendors scrambling as best they can to provide patches and protection to the problem areas, how can enterprises prioritize the efforts of IT administrators so that adequate coverage is provided? The Exploit Effort versus Potential Reward Matrix provides a simple model for thinking about vulnerability triage from the perspective of attackers.

In the first half of 2010, X-Force released alerts and advisories on the vulnerabilities listed in Table 3 which are plotted on a two dimensional chart. The horizontal axis (Exploit Effort to Achieve) represents the effort by the attacker to implement an attack using the vulnerability in question. The vertical axis (Potential Reward) represents the potential for gain that an attacker might achieve.

Many of the vulnerabilities represented by the X-Force alerts and advisories cluster toward the top right-hand quadrant (shaded in red). This quadrant represents issues that provide high payoff for attackers while being relatively easy to implement. These vulnerabilities tend to receive a large amount of exploitation activity on the Internet. In contrast, the one vulnerability represented in the lower left-hand quadrant (shaded in yellow) states that this vulnerability is relatively difficult for the attacker to exploit while providing a minimal potential reward.

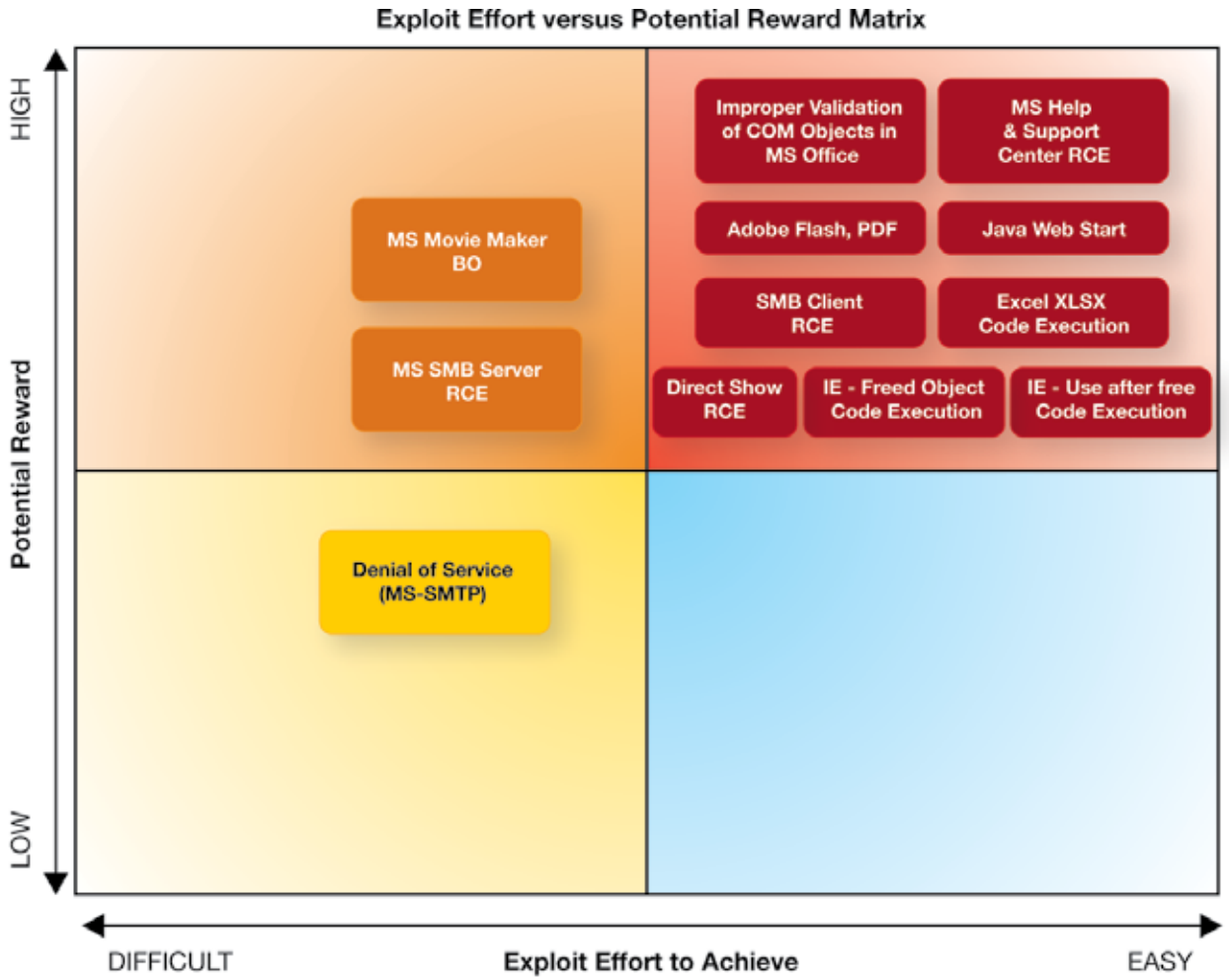


Figure 5: Exploit effort versus potential reward matrix

Later on page 24 of this report, we discuss how early announcement of critical vulnerabilities before a vendor has supplied a patch can impact customer activity and quickly become real issues to handle. We discuss in more detail two of the vulnerabilities that are listed in the upper right hand of this matrix which caused concern. On the other side of the spectrum is a denial of service issue that impacts Microsoft SMTP services. While the threat of a denial of email services is significant for network operators, this sort of attack provides little economic opportunity for attackers. No one ever released a public exploit for this particular vulnerability, so it remains relatively difficult to target.

The Internet Explorer Freed Object Code Execution vulnerability provides an example of the way that vulnerabilities can move across the matrix, from left to right, as more information is disclosed about them. This issue was initially discovered by attackers and used in targeted attacks. The effort to find and exploit a unique, undisclosed, unpatched vulnerability is relatively high. But once the issue was publicly disclosed, exploits were disseminated publicly, and now, the issue is fairly inexpensive for bad guys to target.

Date	Alert/Advisory	Vulnerability Name
14 June 2010	Alert 370	**Microsoft Windows Help and Support Center Could Allow Remote Code Execution** The vulnerability in Microsoft Help Center is due to the injection of invalid unicode characters in a specially crafted hcp request.
20 April 2010	Alert 367	**Java Web Start** A Java feature for launching and installing applications has a design flaw allowing arbitrary commands to be passed directly to the Java Virtual Machine.
08 June 2010	Alert 368	**Improper Validation of COM Objects in Microsoft Office** Microsoft Office applications fail to properly validate COM objects embedded in compound documents. This allows attackers to bypass the security settings of Office and embed known flawed objects in Office files. Upon exploitation of the pre-existing flaws in these controls, attackers can achieve arbitrary code execution.
09 March 2010	Alert 364	**Microsoft Internet Explorer Use-after-free Code Execution** Microsoft Internet Explorer could allow a remote attacker to execute code on the system, caused by an invalid pointer reference error.
07 June 2010	Alert 369	**Flash Player, Adobe Acrobat and Acrobat Reader Remote Code Execution** This vulnerability could result in remote code execution if a victim opens a specially-crafted PDF (portable document format) file or SWF file.

Table continued on page 23

Date	Alert/Advisory	Vulnerability Name
15 January 2010	Alert 359	**Microsoft Internet Explorer Freed Object Code Execution** Web exploit toolkits are notorious for targeting browser and browser-related exploits such as this vulnerability. This vulnerability is reported to have been involved in the high profile attacks on Google and at least 20 other large companies.
13 April 2010	Alert 366	**Microsoft DirectShow Remote Code Execution** This vulnerability is present on all modern Microsoft Windows operating systems. Successful exploitation of this issue would provide an attacker with complete control over the endpoint target. The use of malicious media files like images and movies has been prevalent in the past years.
09 March 2010	Alert 363	**Microsoft Excel XLSX Code Execution** Microsoft Excel could allow a remote attacker to execute arbitrary code on the system, caused by the improper parsing of the Excel spreadsheet file format.
09 February 2010	Alert 360	**Microsoft Windows SMB Client Remote Code Execution** This vulnerability is in a core component of most modern Microsoft Windows operating systems, including Windows 7. The easiest attack vector requires an attacker to set up an SMB server and entice a user to click a link to the server. Successful exploitation provides the attacker with complete control of the end user's system.
09 March 2010	Alert 362	**Microsoft Movie Maker Buffer Overflow** Microsoft Movie Maker is vulnerable to a buffer overflow, caused by improper bounds checking when parsing malicious Movie Maker (.mswmm) files.
09 February 2010	Alert 361	**Microsoft Windows SMB Server Remote Code Execution** This vulnerability is in a core component of most modern Microsoft Windows operating systems, including server editions. If crafted properly, the attack would provide full remote code execution without any end user interaction, although a denial-of-service is more likely to occur. However, the attacker must first have authentication rights to the system, and the guest account would not work in this scenario.
13 April 2010	Alert 365	**Denial of Service Conditions in Microsoft Exchange and Microsoft SMTP Service** Successful exploitation could result in SMTP Service restart and repeated attacks could completely disrupt Microsoft Exchange services. As SMTP services are often exposed to the Internet and email is usually considered a business critical function, the business impact of this vulnerability is more significant than for typical Denial of Service issues.

Table 3: X-Force alerts and advisories, 2010 H1

Public disclosures that had impact

The two most critical vulnerabilities disclosed in the first half of 2010 were remote code execution vulnerabilities in Java Web Start and Microsoft Windows Help and Support Center. Both vulnerabilities were publicly disclosed by researcher Tavis Ormandy before patches were available from the respective vendors. Whenever exploit details are publicly available before a patch, this provides a maximum opportunity for attackers with a minimum amount of effort, and the rapid real world exploitation activity we've seen connected with these vulnerabilities is in line with what our model would predict as shown in Figure 6.

Taking the Java Web Start vulnerability as an example, on April 20, 2010 IBM Security released new signatures to protect our customers and announced this threat on our website. As the data on the right represents, we see immediate effect from the customers who have deployed these new signatures beginning on April 21, 2010. In the first day over 100 security events were seen across the customer base and these numbers continued to climb into the end of June before we start to see a slow decline in the month of July.

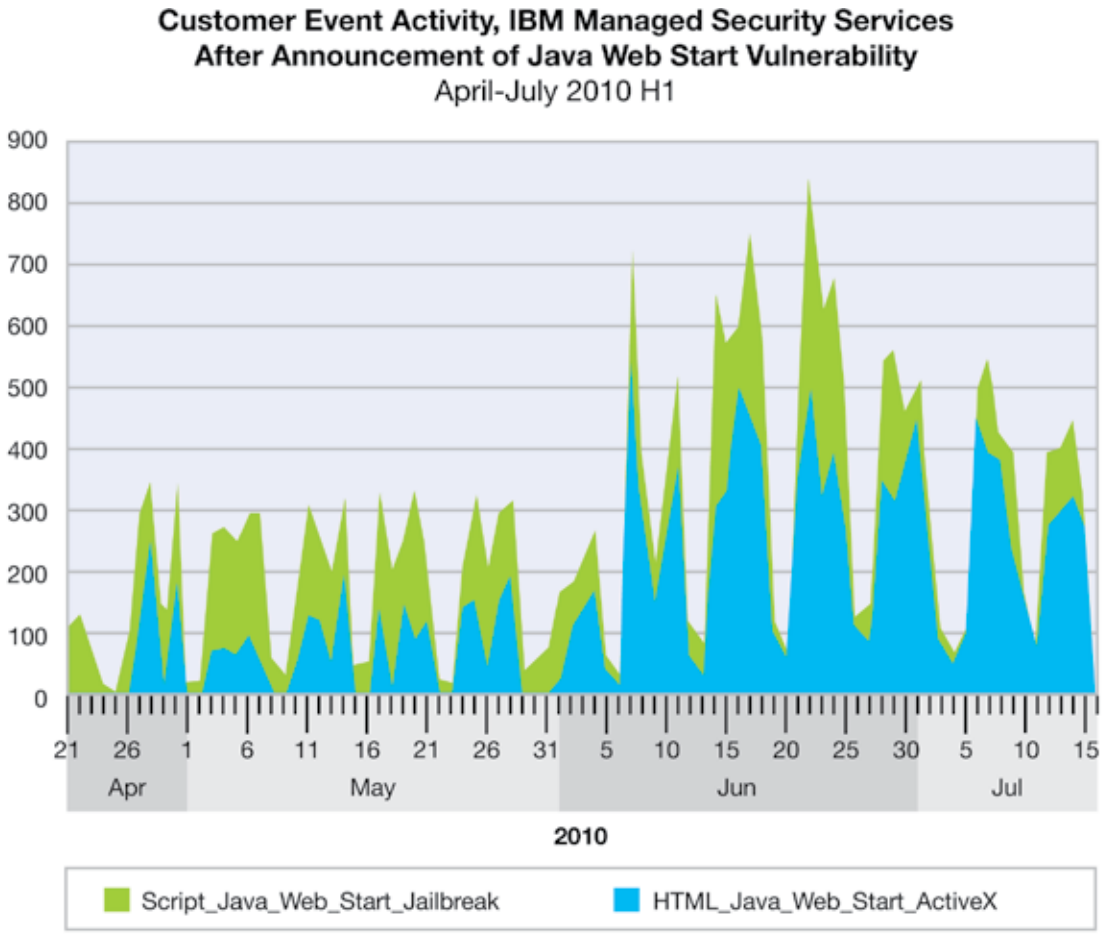


Figure 6: MSS customer event activity after announcement of Java Web Start vulnerability, April-July 2010 H1

Conficker update—what has happened since the end of 2009?

The Conficker worm was one of the biggest computer security stories of the past few years. We decided that an update for this trend report might be in order, but first let's start with a little history.

Conficker first started spreading during the fall of 2008. The initial variant (called Conficker.A) targeted a recently patched remote code execution vulnerability in Microsoft's RPC stack. Conficker.A was not terribly successful relative to historical worm outbreaks that targeted similar vulnerabilities, such as the Blaster worm in 2003. This is largely due to improvements in how rapidly the Internet responds to vulnerability disclosures. By the end of 2008 Conficker had infected only a few hundred thousand hosts.

At the end of December 2008, a new version of Conficker came on the scene, Conficker.B. Conficker.B added a host of alternate propagation vectors to Conficker's arsenal. Conficker.B could spread through USB keys, over file shares, and by cracking bad passwords on Windows domains. These alternate vectors made Conficker more nimble. It could use different vectors to establish footholds in various networks. The consequence was a massive expansion in the number of infected hosts.

By winter 2009, a posse called the Conficker Working Group had formed to deal with Conficker. Conficker.A and B nodes attempt to contact 500 randomly generated domain names every day in search of updates. The Conficker Working Group formed to register all of those domain names so that the Conficker operators could not update the bot. Unfortunately, one update to Conficker did get through. This new variant is called Conficker.C.

Conficker.C expanded the list of domains from 500 to 50,000, and it added an encrypted P2P update mechanism that did not rely on domains that the Conficker Working Group could register. These new features made it impossible for the Conficker Working Group to prevent Conficker.C from updating. Fortunately Conficker.C did not include propagation code, so the infection had no way to spread past the initial nodes.

The Blaster worm started propagating on the Internet in August of 2003. It exploited a vulnerability (MS03-026) in the Microsoft Windows RPC stack that was very similar to the vulnerability exploited by Conficker.A. Blaster reached peak propagation within eight hours of its initial release and ultimately infected between eight and sixteen million hosts on the Internet. Blaster launched a distributed denial-of-service attack against WindowsUpdate.com but the impact was minimal because Microsoft actually used a different address for hosting updates.

X-Force response to Conficker

X-Force researchers reverse engineered the Conficker code and developed signatures in our IPS products that can detect and block Conficker.C P2P (peer-to-peer) traffic. Figure 7 shows those traffic levels slowly deteriorating over time. However, when pulling our latest data we noticed a slight uptick in June activity as we were heading to the press with this report. X-Force researchers will continue to investigate why there is a possible change in this activity and keep our readers updated by the **Frequency X blog** once we understand more.

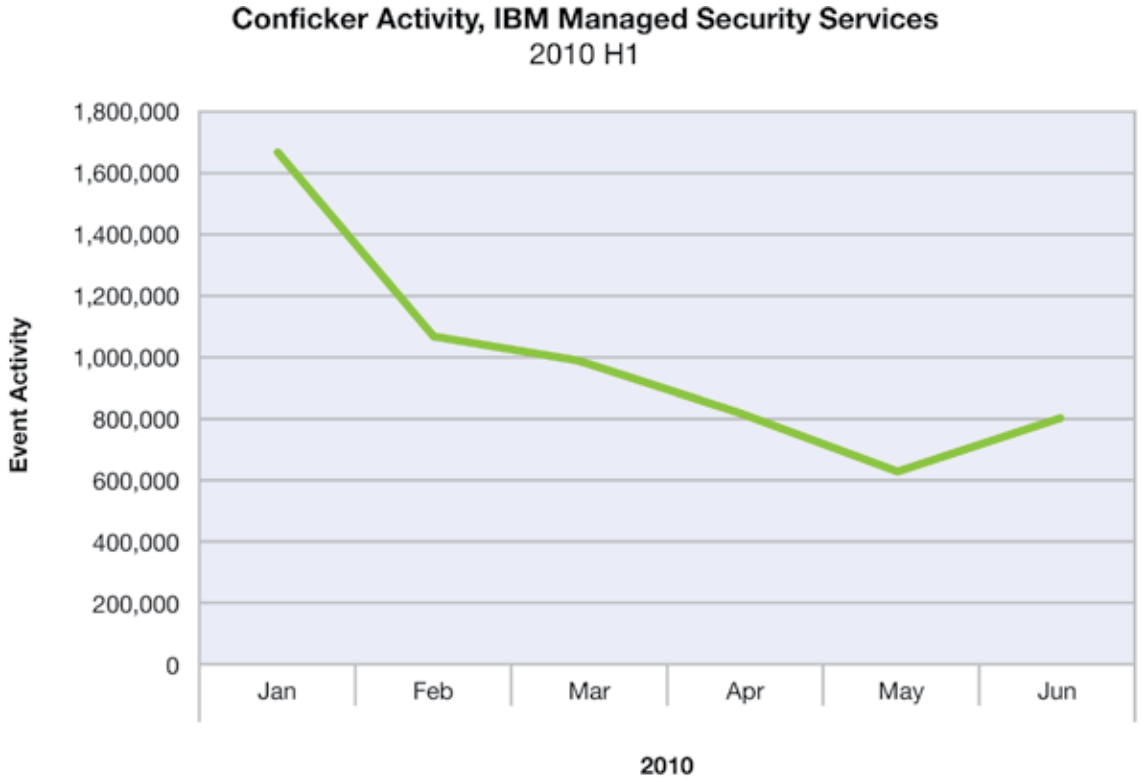


Figure 7: Conficker activity, IBM Managed Security Services, 2010 H1

This is consistent with Conficker.C data from X-Force's Darknet (Figure 8 below.)

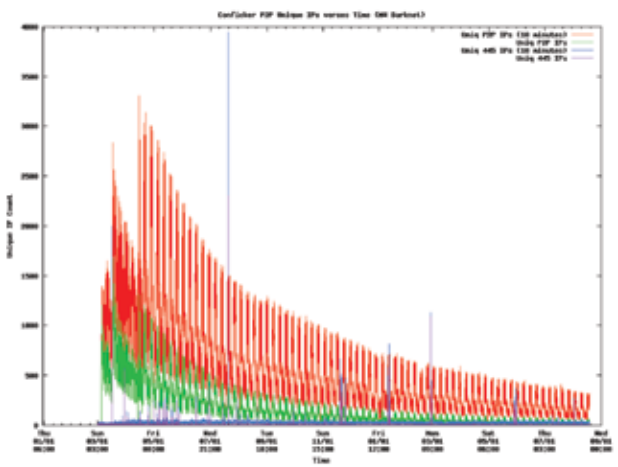


Figure 8: Conficker Working Group

As well as the Conficker Working Group's Sinkholes (Figure 9)

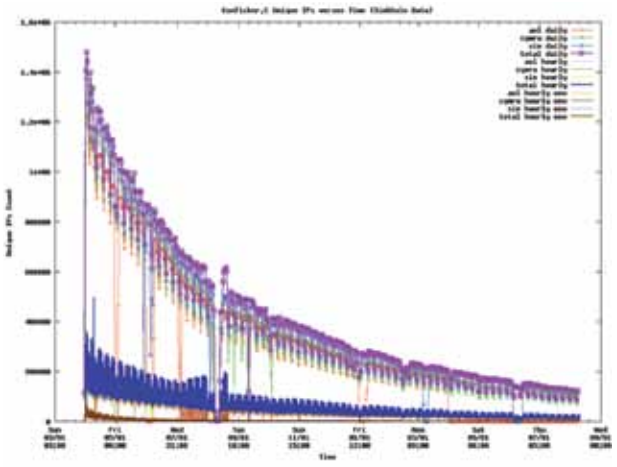


Figure 9: Conficker Working Group

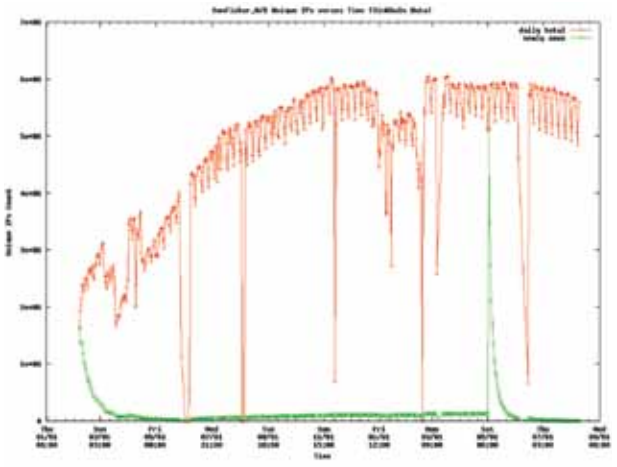


Figure 10: Conficker Working Group

Conficker.C is slowly disappearing, as infected nodes are cleaned up with anti-virus software or simply break down and are removed from the Internet. As Conficker.C has no way to infect new nodes, it has no way to maintain its population on the Internet. This is a good thing, as the Conficker Working Group cannot prevent the botnet operators from updating Conficker.C nodes. There are almost 200,000 Conficker.C nodes still out there on the Internet, waiting for an update. So far no widespread update has been sent out.

The Conficker.A/B botnet is much larger, comprising between 5 and 6 million nodes according to the Conficker Working Group (Figure 10 above). This botnet topped out at this size around November of 2009 and has managed to hold steady for nearly 9 months. We imagine that every day some Conficker.A/B nodes die, for the same reasons that Conficker.C nodes die—anti-virus installations and system failures. However, Conficker.A/B nodes still propagate by breaking into new systems. In order to maintain its population, the rate of infection of new nodes must be the same as the rate of node death. It is interesting that these rates have become so stable over such a long period of time.

The future of Conficker?

Fortunately, there is nothing that the creators of Conficker can do with their five to six million node Conficker.A/B botnet, because the Conficker Working Group is still registering the 500 domain names those nodes attempt to contact, every single day. If the Conficker Working Group abandoned its efforts, the attackers would be left with control of a very substantial botnet that would pose a significant threat to the Internet infrastructure.

Even though it has been nearly two years since Conficker started to propagate, it is not dead. It remains a slumbering pair of dragons, only one of which is contained. Most of the lessons that have been learned from this experience are not good. Clearly, worms can still be used to build very large botnets on the modern Internet. Clearly, those botnets can persist for years with a very large numbers of nodes which could be put to various malicious purposes. And in the case of Conficker.C, clearly it is possible to build botnet command and control systems which cannot be globally mitigated.

However, the Conficker experience and the formation of the Conficker Working Group have connected infrastructure operators and security companies into a tighter mesh. When the next major worm outbreak occurs, this community will be ready to respond.

Trending in the dark—what does malicious traffic look like?

There are many data resources at IBM analysts' fingertips to utilize for the purposes of trending. One of those resources is a darknet, also known as a black-hole network. This space is continuously monitored and all incoming traffic is captured in its entirety and stored for analysis and long term archiving. With an aperture of 25,600 addresses, this darknet is part of a larger information gathering network. By the very nature of a darknet, no packets ever originate from these addresses and no legitimate traffic would ever be destined to these addresses. Additionally, these addresses were never allocated to any active legitimate device or service on the Internet. They are, nonetheless, advertised as a part of a legitimate "/16" network and are fully routable from the greater Internet. All traffic into this network may therefore be assumed to be malicious.

Spoofed denial of service attacks

Looking at the data over the past several years, a couple of interesting patterns begin to emerge. The first trend is the gradual rise in backscatter activity (Figure 11). Backscatter is actually a side-effect of a spoofed Denial of Service (DoS) attack. By spoofing the source address in Internet Protocol (IP) packets sent to a victim, the victim's system is unable to distinguish between the spoofed packets and legitimate packets and responds to the spoofed packets. These response packets are known as backscatter.

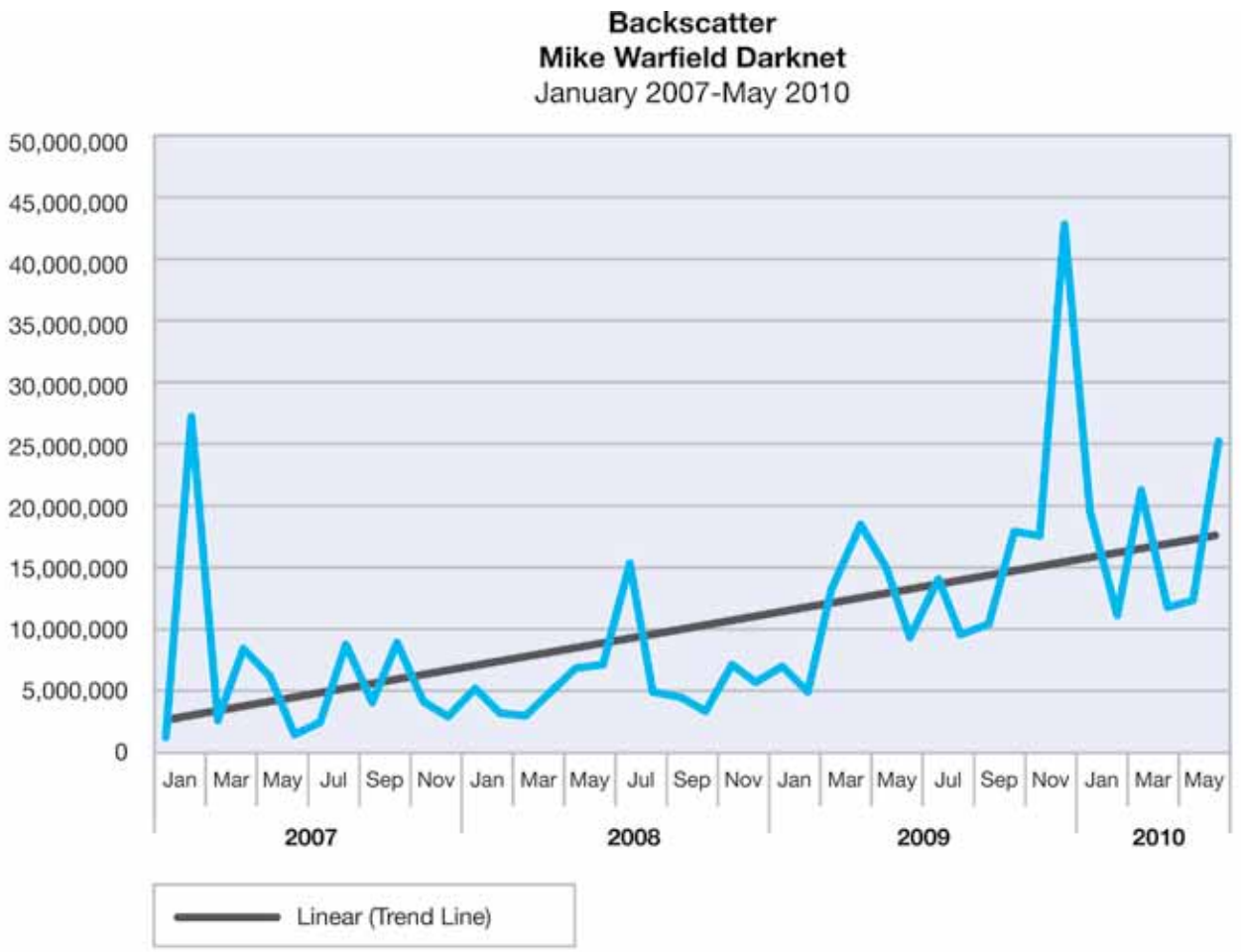


Figure 11: Backscatter, Mike Warfield Darknet, January 2007-May 2010

In Mike Warfield's darknet, each SYN-ACK backscatter packet received is an indicator that an attacker sent a spoofed packet to a well known service port on the machine under attack spoofed from one of Mike Warfield's darknet addresses.

While there has been a gradual increase in backscatter activity since 2007, there was actually a large jump year over year between 2008 and 2009. Part of this increase is due to a significant spike in activity in 2009—the largest in the three and half year period. This trend of higher than previous year averages continues in 2010. At the close of Q2, the average count for the first half of 2010 is just slightly higher than the total average for 2009. When you look at Figure 12 you are actually reviewing the increase in volume from 2007 through 2010 of spoofed denial of service attacks on the Internet.

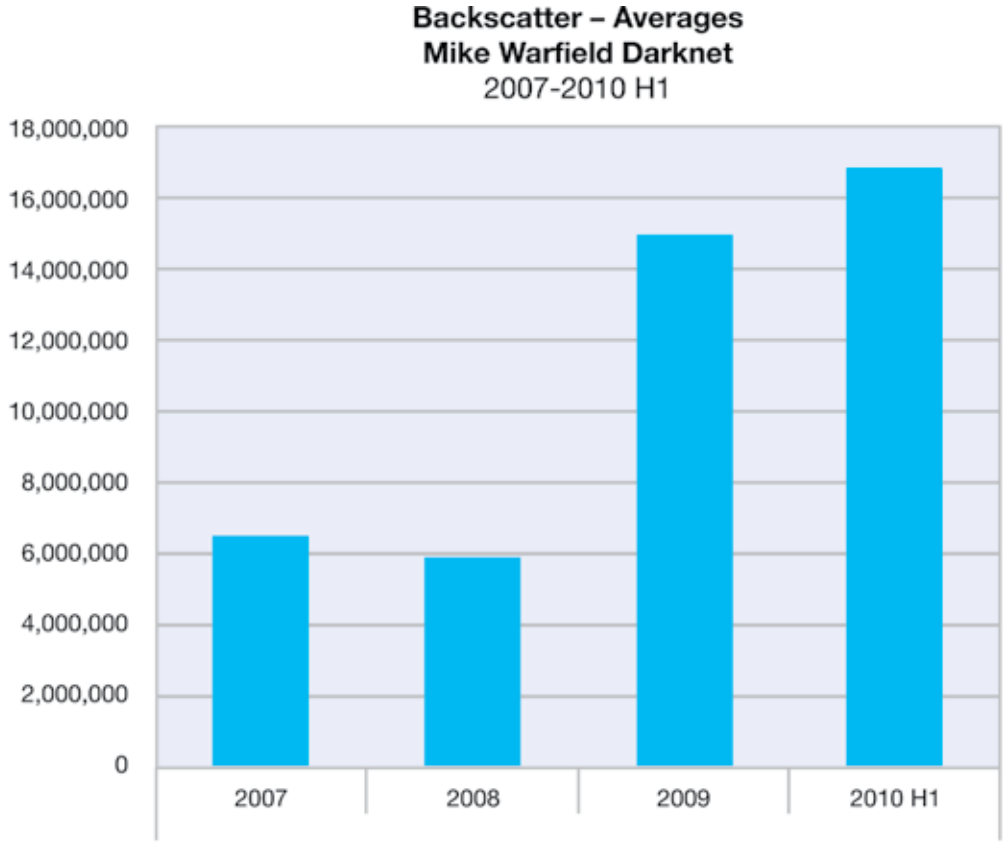


Figure 12: Backscatter – Averages, Mike Warfield Darknet, 2007-2010 H1

What can we deduce from this gradual rise in backscatter data and, in some instances, large jumps of backscatter activity? Since the majority of the backscatter data results from DoS attacks, we are able to speculate that there has been a steady increase in spoofed DoS attacks since 2007. However, backscatter is subject to some high degree of variability due to the nature of what is being collected and what is occurring. Some intense periods of backscatter are the result of internecine warfare within and between the various attacker camps. During this warfare, one group attempts to block or take over the resources of another group. This "shelling match" between warring camps can result in a sudden increase in backscatter traffic and backscatter source addresses. It generally ceases as suddenly as it initiates. This type of activity most likely contributed to the dramatic spikes in February 2007 and December 2009 as seen in Figure 11.

Brute force attacks

Mike Warfield's darknet also provides us with insight into the world of brute force attacks. A brute force attack, in the computer security sense, involves an attacker trying to gain unauthorized access to a system by trying a large number of password possibilities. Some of the services that are often targeted by brute force attempts are: SSH (TCP port 22), Telnet (TCP port 23), RealVNC (TCP port 5900), and Microsoft Remote Desktop (TCP port 3389).

Figure 13 compares the average activity of these ports since 2008. Activities on both RealVNC and Microsoft Remote Desktop ports show a slow growing upward trend. In contrast, the SSH shows a slow and steady decline and Telnet has taken a sharp decline since 2009.

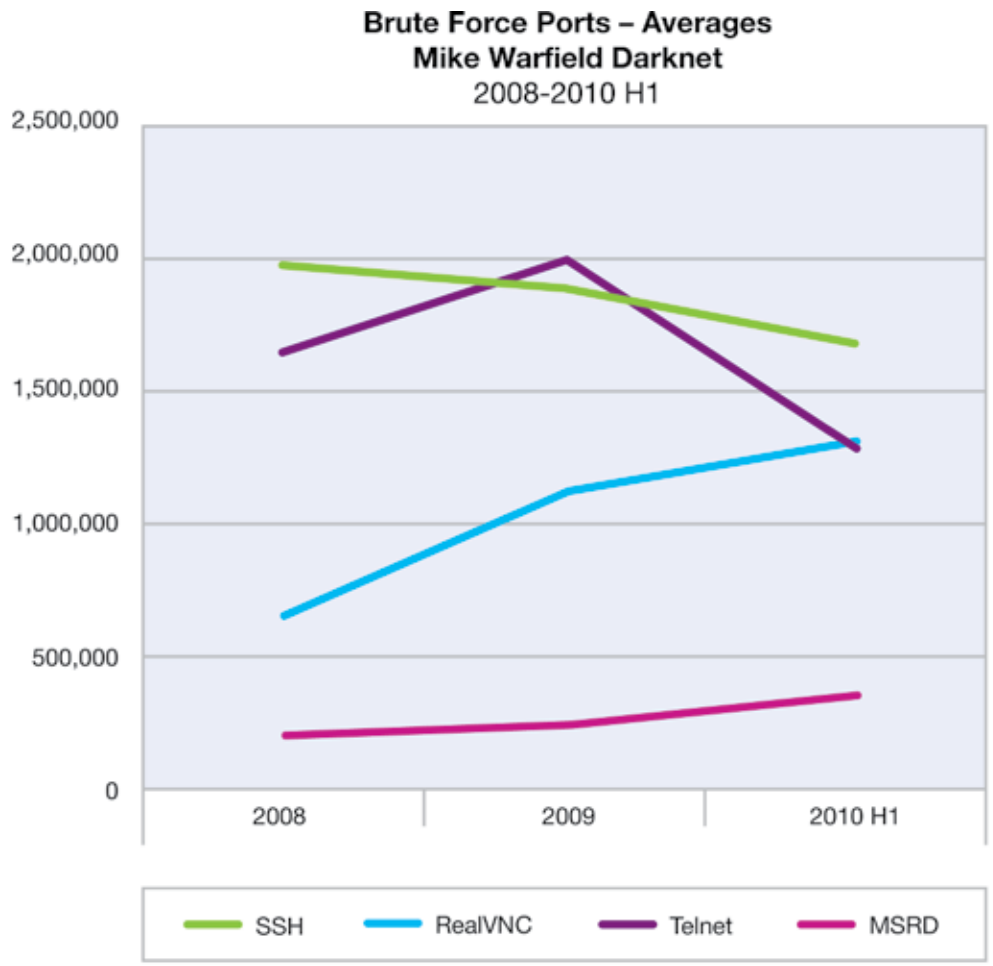


Figure 13: Brute Force Ports – Averages, Mike Warfield Darknet, 2008-2010 H1

This data indicates that there may be less interest in targeting the SSH and Telnet ports over the past year and half, whereas the RealVNC and MS Remote Desktop ports are increasing in popularity. Does this somehow correlate to when vulnerabilities targeting these protocols are released? Figure 14 shows the total darknet activity with these four ports over the past two and half years. Some of the increases may be related to vulnerability disclosures. For instance, a vulnerability targeting RealVNC in early May of this year may have contributed to the upward tick seen at the end of the second quarter. Six vulnerabilities targeting SSH were released in May 2008 and a significant increase is observed during this month on the darknet chart. A larger increase in SSH activity occurred in December of 2008 the same month a vulnerability targeting FreeSSHd was disclosed.

However, all the significant peaks in the data cannot be explained away with a vulnerability disclosure occurring around the same timeframe. In fact, the last publicly disclosed vulnerability affecting Telnet was disclosed in March 2005. Additionally, some of the most significant increases in activity, such as the spike in RealVNC in August 2009, did not occur anywhere near the same time as a vulnerability disclosure. This is an indication that attackers are not always using the latest vulnerability, rather they often rely on older vulnerabilities to carry out their exploitation.

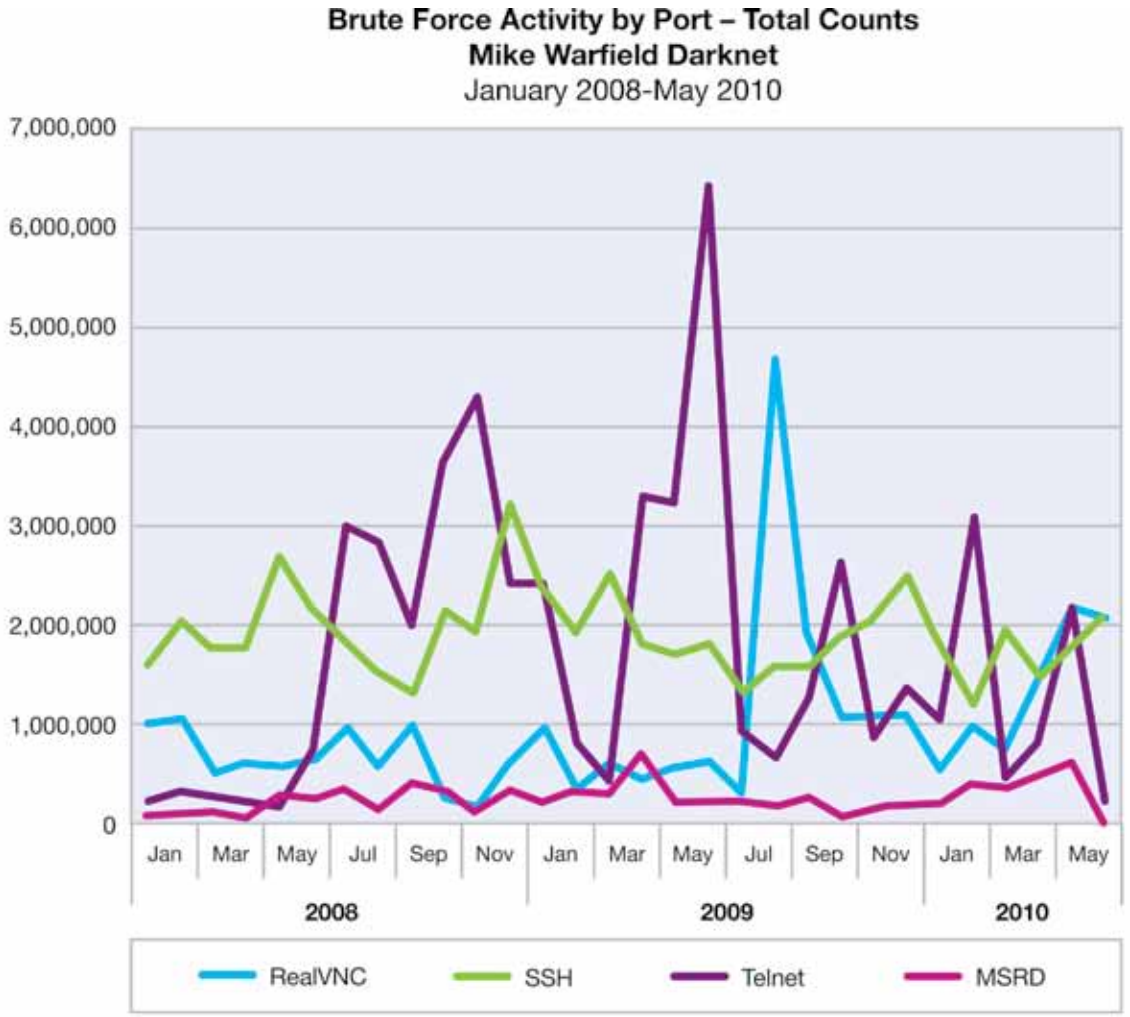


Figure 14: Brute force activity by port – total counts, Mike Warfield Darknet, January 2008-May 2010

Computer Crime—who's tricking who?

Zeus botnet—facts, myths and understanding how these botnets operate

The Zeus/Zbot family of botnets has been plaguing the Internet for years now. The threat has been evolving and new versions and new capabilities are always being discovered. According to media reports, Zeus has been responsible for millions of dollars of losses to companies and individuals due to theft of personal information.

Zeus botnet operators usually infect new PC's by either mass emailing malicious documents to victims, or directing the victims to a website that serves malicious content that installs a Zeus bot. Once installed, Zeus will monitor Internet traffic on the infected PC and report information back to a central Command and Control (C&C) sever. The information collected depends on how the operator configures the bot, but in many cases it collects bank account information. This information is collected regardless of security and encryption settings on the PC—Zeus can inject code directly into the Web browser to collect personal information.

Once the victim's information is collected, it is either used directly by the botnet operator or sold to other criminal groups online.

Myths about Zeus

There are many myths about Zeus and how it operates. Some of these are spread by mass media outlets and even the more technologically-knowledgeable IT media outlets. Many times these myths and misconceptions arise because of misuse of terminology, and some people will argue that the semantics are unimportant, but on the X-Force we believe in sticking to rigid definitions of malware terms in order to accurately describe threats.

There is a single Zeus botnet

This is false. The Zeus Builder toolkit sold online allows anyone to create and manage their own Zeus-based botnet. There are hundreds, or even thousands, of separate Zeus botnets active at any given time. Zeus Tracker (**https://zeustracker.abuse.ch/**), a sevice of the abuse.ch Swiss security blog, monitors active Zeus command and control servers. At the time of writing, there were 644 active Zeus C&C servers being tracked—each one possibly run by a different group or individual.

Zeus is a virus or worm

False. The traditional definition of virus is a program that spreads and infects machines in a way that requires some user interaction: inserting a floppy disk or USB key, running a program, opening an email attachment, etc. A worm is like a virus but spreads without user interaction—worms will commonly exploit security vulnerabilities to do this. Zeus fits neither of those definitions. It has no capability to spread on its own. It is more accurately defined as a backdoor (provides access to a user's computer) or trojan (something other than what it's claimed to be). When people say things like "Zeus is spreading", it can lead one to believe that Zeus has the ability to spread on its own, but this is not the case.

Zeus uses vulnerabilities and exploits to install itself

This is also false. Zeus itself is just a backdoor or Trojan. However, many groups and individuals that use Zeus to steal information will deliver it using an exploit. In this case, Zeus is the payload of the exploit but the payload itself has nothing to do with the exploit. We have seen many vulnerabilities used to deliver Zeus—PDF exploits, a variety Web-based ActiveX control exploits, etc. Every time a new vulnerability is publicly disclosed, someone will use it to deliver Zeus. This has nothing at all to do with Zeus itself or the creators of Zeus Builder—Zeus is just a very effective payload if the goal is stealing financial information from victims.

New version of the Zeus botnet toolkit

Early 2010 saw the release of an updated version of the Zeus botnet kit, dubbed Zeus 2.0. The major new feature included in the new version was support for intercepting personal data from the Firefox Web browser—older versions of Zeus only included the ability to intercept data from Internet Explorer.

There are many other changes from earlier versions of Zeus.

Changes in Zeus 2

These are just some of the changes that were made in Zeus 2. Many of the changes were made to allow Zeus to more effectively infect machines in an enterprise environment where users may not have Administrative access to their computers.

Auto-start technique – The older version of Zeus would install itself to run automatically at system start by utilizing the `HKLM\Microsoft\Windows NT\CurrentVersion\Winlogon` registry key when the infected user had Administrator privileges, and `HKCU\Microsoft\Windows\ CurrentVersion\Run` when executed without Administrator privileges. Removal was difficult because the Zeus bot would continually monitor the key and prevent any modification. Booting Windows in Safe Mode would still let Zeus load when the `Winlogon` key was used. In the new version of Zeus, the `HKCU\Software\Microsoft\ Windows\CurrentVersion\Run` is used to make Zeus auto-start regardless of the user's privilege level. It's easier to detect and remove this entry from the registry. This also causes Zeus to only run for the user that was initially infected.

File location – The older version of Zeus will place a copy of itself, unmodified, in the `Windows\ System32` directory if the user had Administrator rights, or in the user's `Application Data` directory if not. The file was usually named `sdra64.exe`. The new version will place a copy of itself in a randomly named subdirectory in the user's `Application Settings` directory with a random name.

Network traffic – The protocol used to communicate with the Command and Control (C&C) server is largely the same from the network level. HTTP POST data is encrypted with RC4. One visible change to Zeus's HTTP requests is that it now uses the "`Cache-control: no-cache`" directive in the HTTP request headers instead of the older HTTP-1.0 style header containing "`Pragma: no-cache`".

Unique infection binaries – Zeus makes slight and random modifications to the copy of itself that it drops in the user's Application Settings directory. This means that if a single Zeus installer is sent to many people, each resulting infection will contain a slightly different executable. Only a small number of bytes are changed, but it's enough to give the file a different SHA or MD5 hash. The file size can also be different.

Binaries bound to a specific machine – Zeus now uses a technique similar to commercial software copy protection to make analysis of the installed executable harder. Once a machine has been infected, the original executable is removed and the file stored on disk won't run on a different computer. It does this by checking the boot drive's volume GUID and the directory the executable is stored in. If this information doesn't match what's stored in the EXE itself, Zeus won't run. This means that common auto-analysis techniques won't work.

Configuration file location – In Zeus 1.3 and previous versions, the configuration file was downloaded from a server and stored as "`local.ds`" in a hidden directory within `Windows\System32` (or the `Application Data` directory when the user doesn't have Administrator rights). Zeus now downloads the file and stores it as a random name in the user's `Application Settings` directory. Because the file has a random name, it's more difficult to detect than the static named used in older versions.

OS Support – Zeus 2.0 now runs on Vista and Windows 7. Old versions would just crash when attempting to run, but new versions are able to successfully operate on the latest desktop OS from Microsoft. Zeus 2.0 will also work on 64-bit versions of these OS's.

Initial infection vector – There is little difference between the ways that the old and new versions of Zeus are distributed. Infection methods are opportunistic—when a new vulnerability is discovered, cyber criminals will attempt to use it to expand their existing Zeus botnets with new infections. Since Zeus is sold as a botnet creation kit in underground forums, it's used by many different groups and individuals and each one can use a different method to distribute it. Some methods we've seen this year are emails with .zip file attachments containing a Zeus bot, emails with links to .zip or .exe files, emails with links to sites containing exploit packs that will install Zeus, and .pdf attachments using the /Launch exploit and other vulnerabilities. This is by no means an exhaustive list—people will continue using more creative methods to get any malware, including Zeus, installed on as many machines as possible to earn money.

Protecting yourself from Zeus

There aren't any special steps to take to protect against Zeus specifically. Safe Internet habits will protect computer users from infection by any malware.

PC safety

- Run as a non-administrator user. Although Zeus can still infect your PC, the damage that can be done will be minimized and the infection is easier to clean.
- Keep your computer updated with the latest patches. Doing this will limit the amount of malware that can run on your PC, as well as limit the available attack vectors. Pay particular attention to updates for your operating system, office and document software, and Web browsers and plugins.
- Install Anti-Virus software and keep it up-to-date. While AV software won't protect against all malware threats, it does help.

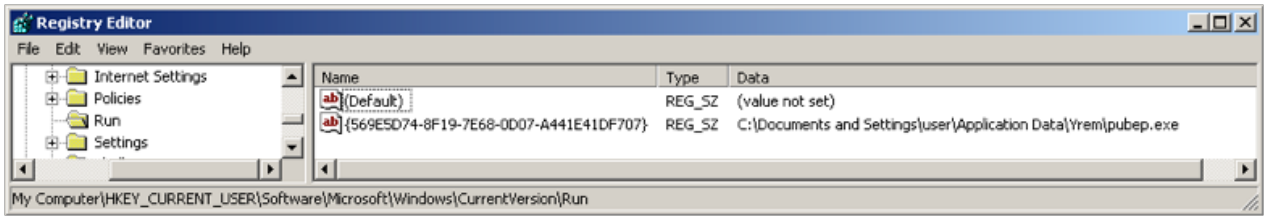
Email and messaging safety

- Be very wary of email attachments. If the attachment is from someone you know, confirm that they are the original sender. Check that the email is from their usual email address.
- Also be wary of links in emails. Many phishing attacks use legitimate-looking emails which contain links to malicious sites. If you get an email from your bank, use a browser bookmark to go directly to the bank's site to log in. This same advice is relevant for messages received from Instant Messenger services and social networking sites.

Indicators of infection

There are a few indicators to look for when a Zeus infection is suspected:

An entry in the HKCU\Software\Microsoft\Windows\CurrentVersion\Run registry key with a GUID-formatted name that points to a file in the user's Application Data directory. Here is an example of what this looks like:



This registry key is constantly being written to undo any attempts at deletion. Deleting the key results in it being immediately replaced. You can use Microsoft's Process Monitor tool to identify this behavior. In this example, the Explorer.exe process is constantly re-writing this registry value:

11:15:24.5760668 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:24.7791862 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:24.9823505 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:25.1858954 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:25.3885276 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:25.5918844 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:25.7948521 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:26.0076655 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:26.2867116 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:26.4958915 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:26.8151519 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:27.0135168 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}
11:15:27.2597238 AM	Explorer.EXE	1272	RegSetValue	HKCU\Software\Microsoft\Windows\CurrentVersion\Run\{569E5D74-8F19-7E68-0D07-A441E41DF707}

Rebooting the computer in Safe Mode and then deleting both the registry entry and the EXE file itself removes Zeus, but we still recommend reinstalling or re-imaging machines that have been compromised with malware. There's no telling what other malware could be lurking on the computer, and removing Zeus may only eliminate one part of an installed malware ecosystem.

BlackHat search engine poisoning

BlackHat search engine poisoning is a technique originally used by spammers to get their search results near the top of search engines so they can earn advertising revenue. Lately, other cyber criminals have been using these techniques to spread malware infections. They are often able to exploit major news events to get malicious links at the top of search result pages on many search engines.

To accomplish this, cyber criminals monitor trending topics on search engines and social network sites. When a new topic is rising rapidly—for example during a major news event—the attackers use standard SEO (Search Engine Optimization) techniques to get their links for those searches to the top of the results page. Because this process is largely automated, sometimes these malicious links appear at the top of search engine results before there's much real news about the major event.

The malicious links themselves are usually cloaked in a few rounds of obfuscation. Many times the links are to PHP pages that contain the search terms in the title (an SEO technique), and the code that produces the page checks the HTTP Referrer to ensure that it came from a valid search engine. This is done to deter Web crawlers and malware analysts. Once it has been verified that the Web browser came from the search engine, other redirection techniques are used. There can be obfuscated and jumbled JavaScript code, embedded Adobe Flash files, or even PDF documents with links to other pages. This is done to make the links hard to follow using automated tools such as the malware crawlers that so many antivirus companies use. After up to five or more levels of redirection, the user's browser ends up on a page containing an exploit toolkit that checks the browser version and available plug-ins and then delivers a malicious payload. At other times, the Web page contains a fake warning about fake viruses discovered on the user's PC, imploring them to install a fraudware rogue antivirus product.

To protect themselves from these kinds of threats, Web surfers should be wary of links they click on in search results. If you're searching for something specific and end up on a rogue antivirus page, do not install the software. If the domain name of the link is totally unrelated to what you're looking for, don't click it. We have seen many legitimate websites compromised by hackers and then used for BlackHat SEO campaigns.

Rogue anti-virus software

Rogue AV, Fake AV, and fraudware. There are many different names that refer to the same piece of software—something that purports to be an antivirus solution that actually does nothing. These products pretend to scan your hard drive and pretend to discover malware, and they ask for your credit card information so you can pay $60 or more to remove the discovered viruses. Of course, once you pay, the only thing that happens is the rogue AV software stops reporting fake viruses.

Rogue AV software has been around for several years. What's new in 2009 and 2010 is that they're using BlackHat SEO techniques to distribute them. It's easy to do a Web search for anything, click on a link, and end up on a page informing you that there is a virus infecting your PC.

If you choose to download and run the software, your PC becomes unusable. Every few seconds another fake warning appears about a fake virus. Popup bubbles appear in your task bar. Web surfing is impossible until you either remove the rogue AV software or pay the fee.

Spam—impersonators of the Internet

Spammers' domains move from .cn to .ru

The following table shows the five most frequently used top level domains used in spam by month. In this table we only consider URLs that actually host spam content.

Rank	January 2010	February 2010	March 2010	April 2010	May 2010	June 2010
1.	com	ru (Russia)	ru (Russia)	com	ru (Russia)	ru (Russia)
2.	cn (China)	com	com	ru (Russia)	com	com
3.	net	net	net	net	de (Germany)	de (Germany)
4.	ru (Russia)	cn (China)	cn (China)	de (Germany)	net	net
5.	info	info	biz	cn (China)	org	org

Table 4: Most common top level domains with real spam content, 2010 H1

The perhaps surprising question is: What happened to China (**.cn**)? Starting with rank 2 in January the rank decreased from month to month. In June 2010, China was ranked at 75. This becomes even more perplexing when reviewing the data for previous years.

In the last few years, Chinese domains (**.cn**) have been the favorite domains of spammers. However, since China has tightened the rules on registering a **.cn** (see **http://www.cnnic.net.cn/html/Dir/2009/12/12/5750.htm**) domain as of mid-December 2009, this seems to have deterred the spammers and moved them in new directions. Before the Chinese NIC closed the doors, it would appear that spammers continued utilizing the pool of already registered domains. After six weeks the pool apparently became empty. Then activity moved from China to Russia. The following chart shows the monthly Top Level Domains (TLDs) used by spammers in the past 18 months.

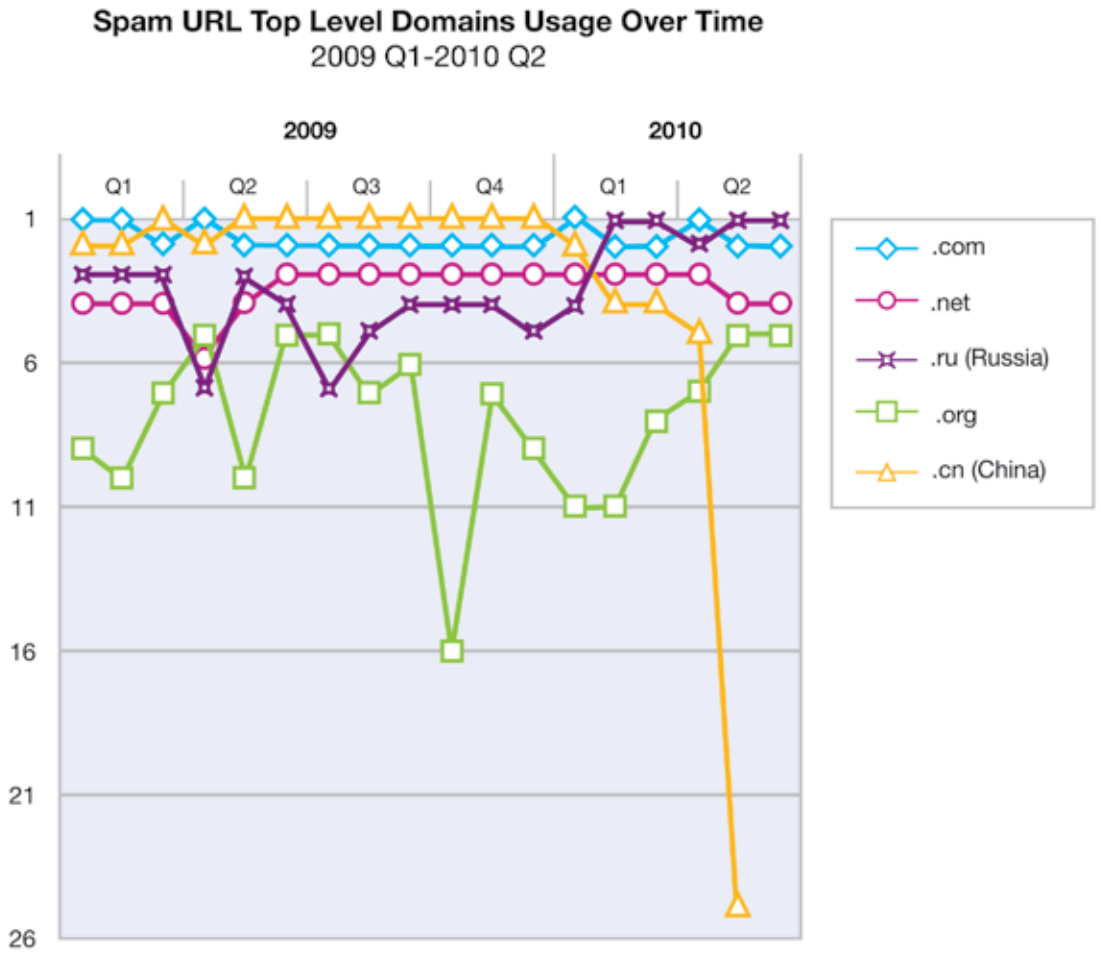


Figure 15: Spam URL top level domains usage over time, 2009 Q1-2010 Q2

On April 1st, 2010, the Russian NIC also tightened their rules to register new domains (see **http://www.nic.ru/dns/service/en/faq_identification.html#q9** for details). However, spammers continue to choose **.ru** domains to provide their offers. In June, 2010, **.ru** is still the topmost used spam top level domain. It will be interesting to see how long this occurs. But what comes then? Do spammers choose another country whose domains are easy to register? Or do they focus on providing their malware via Web hosting services, without the need to register their own domains like other spammers already do?

What can be done to improve these domain registrations?

A good way to prevent the registration of masses of domains for hosting spam content is to request a certificate of registration from companies or a proof of identity for individuals (including document checks). Then people who abuse domains for spam can be identified. China has requested these kinds of certificates since December 2009, and it has been very successful. In Russia a similar new requirement (effective since April 1st) does not appear to have been enforced as carefully to date.

Registration is a legal issue that each country handles differently. It is likely that there will always be some loose registrar out there that provides open doors for spammers. Also, registering domains is only one way to get spam content hosted, another way is to use image hosters or other content hosters, including big players like Google (**googlegroups.com**) or Microsoft (**livefilestore.com**). See the section **Common domains in URL spam**.

Bandwidth irrelevant: byte size of spam significantly increased

The most significant change in the average byte size of spam happened at the end of 2007 and corresponded with the decline of image-based spam. In 2008, byte size began to rise slightly until the McColo takedown later in the year. With the resurgence of image-based spam in summer of 2009, the average size exceeded five kilobytes (KB) for the first time in one and a half years. In the forth quarter of 2009, it decined below four KB.The following chart contrasts the average byte size of spam with the percentage of image-based spam through the end of 2009.

McColo shutdown

After the takedown of the California-based Web hoster McColo in November of 2008, the spam volume dropped to around 25 percent of previous levels. The sudden and extreme volume and country distribution changes observed after the shutdown demonstrated that McColo was the base operator of spam bots all around the world. More details on the McColo takedown and its consequences can be found in the IBM Security 2008 and 2009 X-Force Trend and Risk Reports.

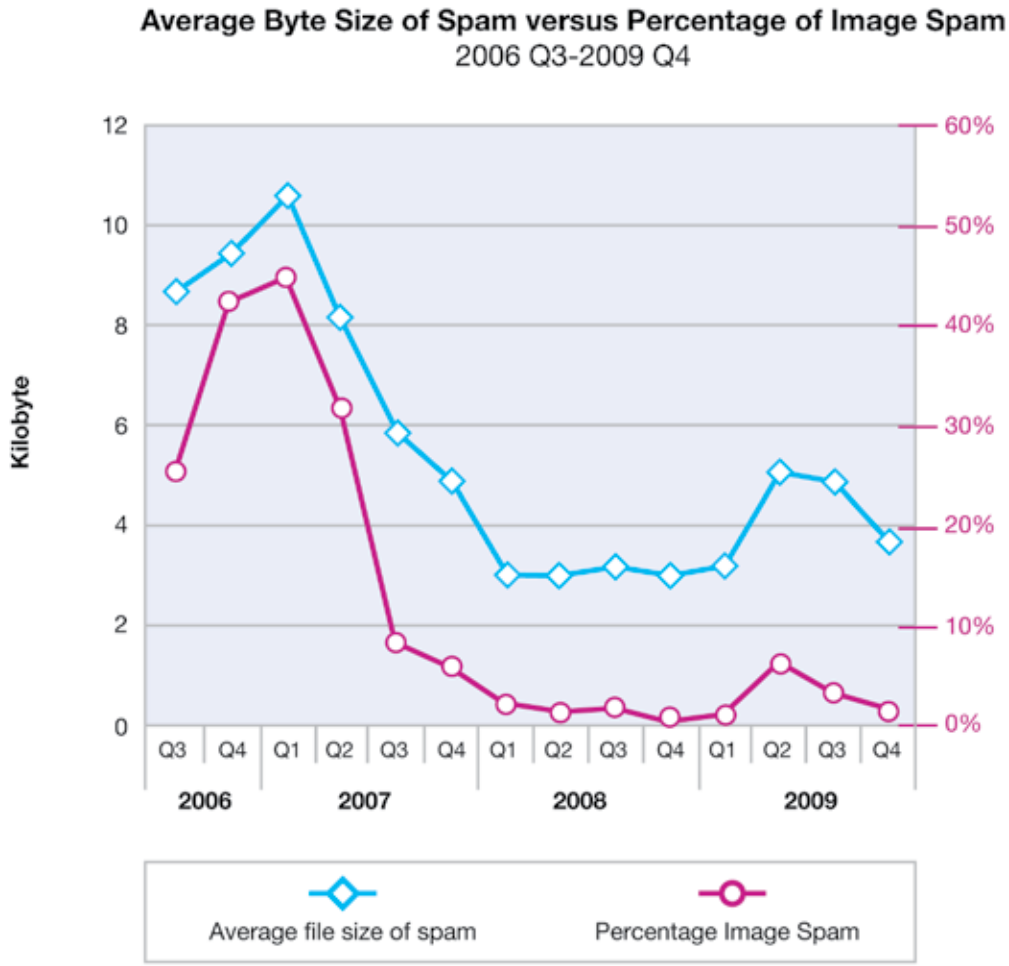


Figure 16: Average byte size of spam versus percentage of image spam, 2006 Q3-2009 Q4

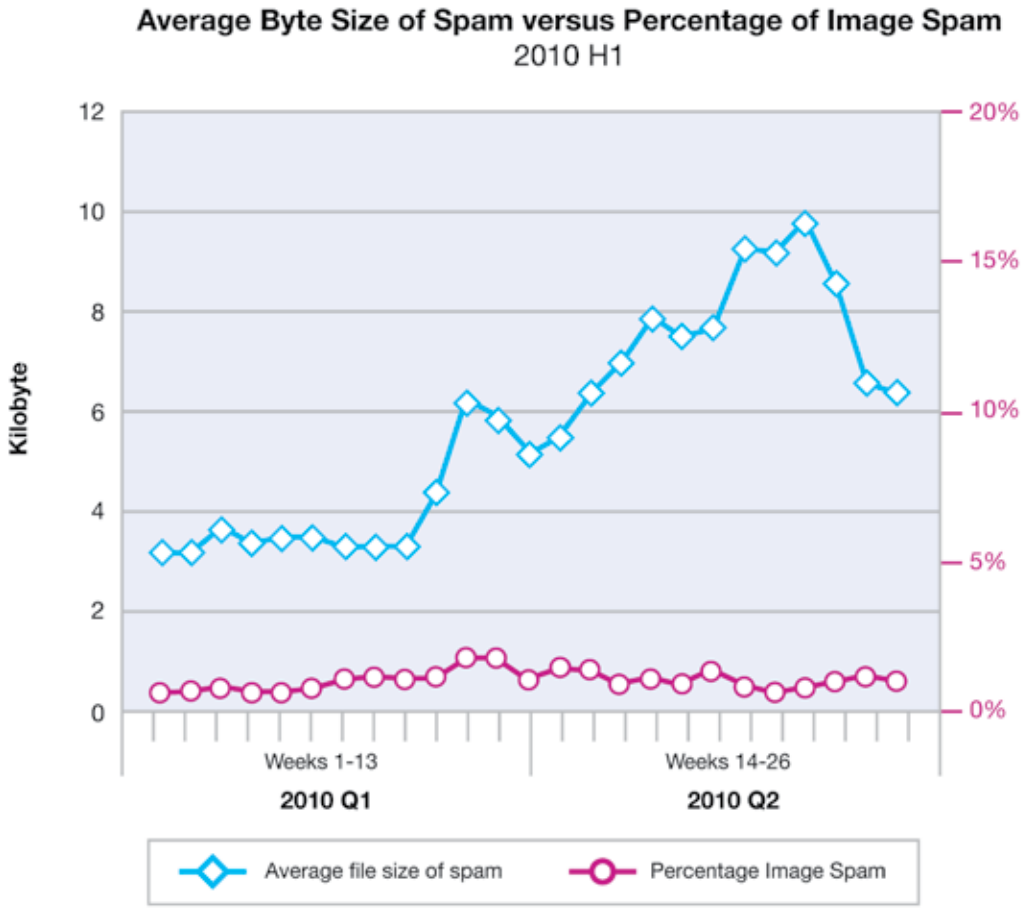


Figure 17: Average byte size of spam versus percentage of image spam, 2010 H1

Both graphs run strictly in parallel. But this changed dramatically since the middle of March. Within a few days, the average size of spam doubled without any changes in the percentage of image-based spam. In the following weeks, the average byte size continued to increase until the beginning of June, reaching an average size of nearly 10 KB. During June, the size declined to about 6.5 KB, still more than twice the amount since the beginning of this random text spam attack. The percentage of image-based spam remained unchanged over that entire period.

When looking at the spam, you can see large text fragments randomly chosen from the Internet. Random text is an old technique that spammers use to make spam look more legitimate—particularly for text-based spam analysis modules. However, recent anti-spam techniques do not have any problems with it. So why do spammers re-activate this old technique? Perhaps they hope that the masses of random text confuse Bayesian classifiers. In particular, self-trained Bayesian classifiers get used in a non-business context, so these spam attacks might be targeted to these non-business users.

You can read more stories and techniques on spam in the current trend section.

Phishing—are you falling for it?

In 2009, financial institutions were unquestionably the dominant target of phishing emails. More than 60 percent of phishing emails were targeted to these institutions. In the first six months of 2010, financial institutions represent about 49.1 percent of those targets. Credit cards represent 27.9 percent, governmental organizations represent 11.2 percent, online payment institutions represent 5.5 percent, and auctions represent 4.6 percent of all phishing email targets. The remaining 1.7 percent of phishing targets consists of other industries such as communication services and online stores.

A new focus on phishing techniques

The percentages as described in **Web Application Threats and Vulnerabilities** on page 71 represent major changes in the distribution of targets within the year and in **Common Domains in URL Spam** on page 94, attackers are focusing more and more on using the good name of trusted websites to lower the guard of end users and attempt to hide their attempts from protection technologies.

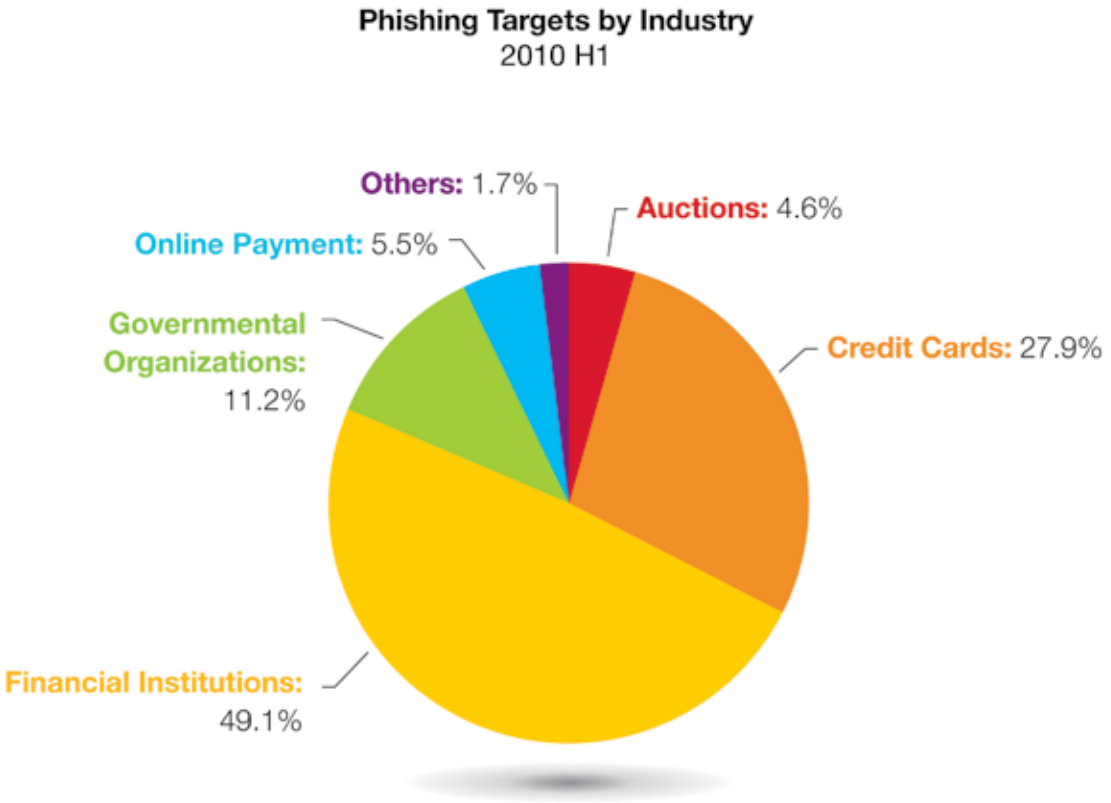


Figure 18: Phishing targets by industry, 2010 H1

Within the last 18 months, financial institutions were the predominant industry targeted by phishing emails. In the first half of 2009, online payment represented a significant portion of phishing emails. However, in the second half of the year, we saw more phishing emails targeting government institutions (predominantly a US tax-related website), credit cards, and auctions. At the same time, the percentage of phishing that targeted online payment organizations declined. In the first quarter of 2010, financial institutions and credit cards declined once more while auctions increased. As we moved into the second quarter of 2010, we began to see all industries declining and phishers once again focusing on financial institutions and credit cards, now representing together more than 96 percent of all phishing emails.

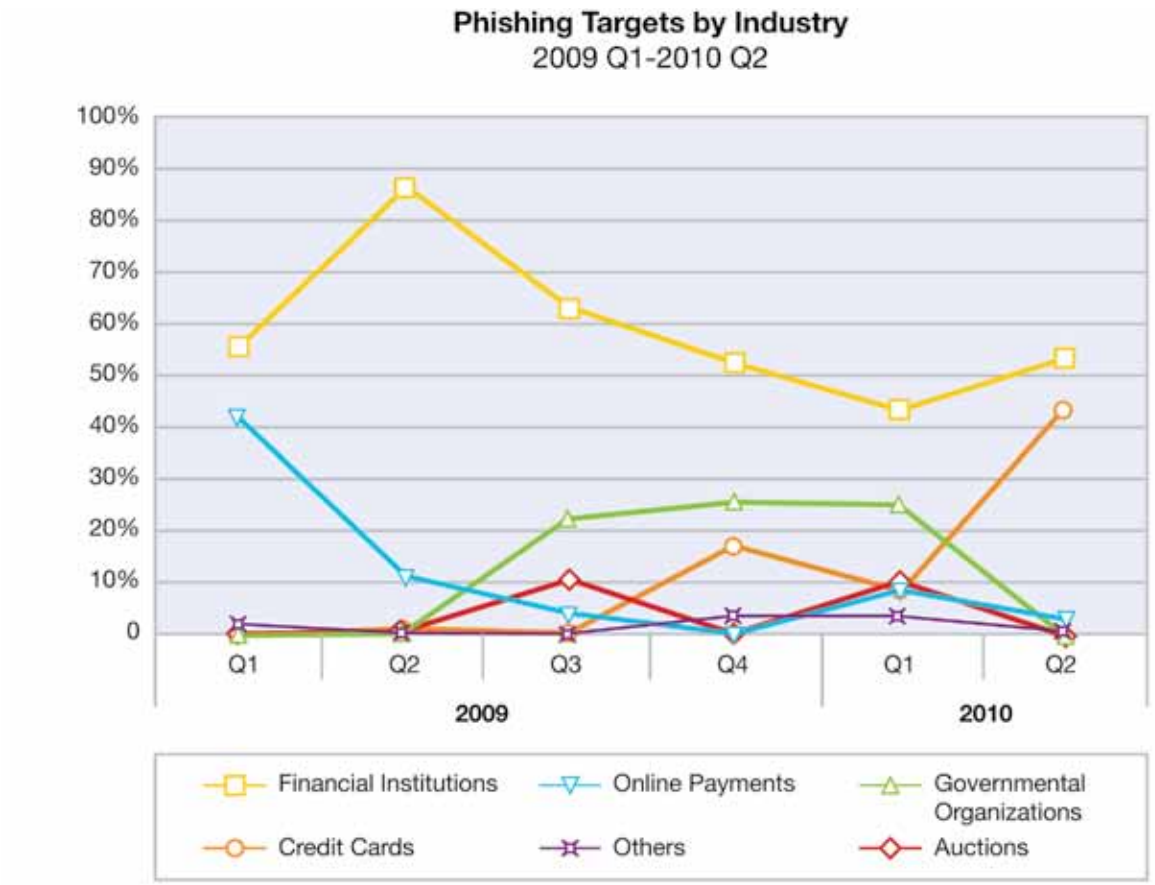


Figure 19: Phishing targets by industry, 2009 Q1-2010 Q2

Why did phishers stop targeting government institutions (in this case a US tax-related website) and now focus on banks and credit cards? One reason may be that after nine months of targeting this tax-related website, the profit is declining and now phishers are focusing on their traditional and proven business to target credit cards and banks.

Financial phishing targeted at banks located in the US

As financial institutions remain a key focus for phishers, it is worth looking closer at the geographies where this activity is prominent. More than two thirds of all financial phishing targets in the first six months of 2010 are located in North America. The remaining 32 percent are targeting Europe.

Figure 20: Financial phishing by geographical location, 2010 H1

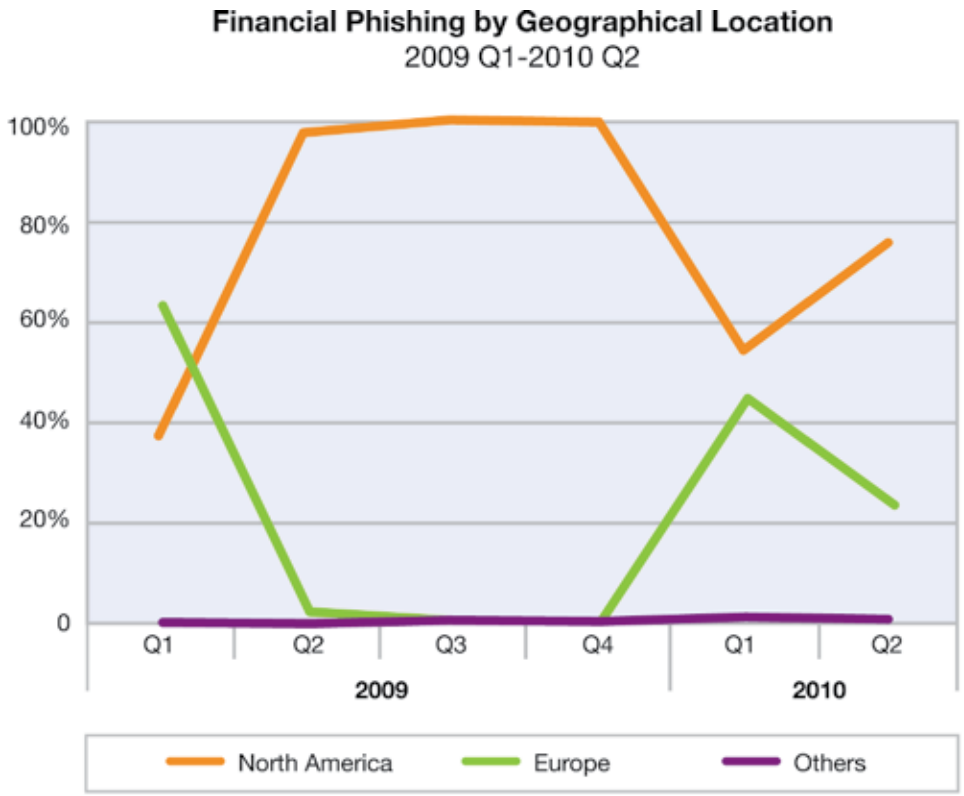


Figure 21: Financial phishing by geographical location, 2009 Q1-2010 Q2

However, after taking a closer look using shorter time frames, more changes become apparent. The following chart shows the shift in geographical location that happened over the course of 2009 and the first half of 2010. While the last three quarters of 2009 were dominated by financial phishing emails targeting US banks (more than 95 percent), in the first quarter of 2010 there were nearly 45 percent financial phishing emails targeting Europe. In the second quarter Europe declined to 24 percent. So why did financial phishers turn towards Europe in the first quarter of 2010 and then turn back again to the US? In the first quarter, the recovery from the financial crisis in Europe became noticeable. While in the second quarter, the budget crisis in Greece and some other European countries lead to the European financial crisis.

We continue discussing the latest phishing trends in a later section of this report.

Future topics—2010 and beyond

IPv6 deployments—we will soon be out of IPv4 addresses; are we ready?

The old generation Internet, IPv4, has continued to explode in terms of addresses and in routing tables. Addresses are now bumping up against the limit of what's available and are projected to run out some time in 2011 at the Internet Assigned Numbers Authority (IANA), and later at the Regional Internet Registries (RIRs). The end of that address road, though, is not a hard stop but a soft landing at the end with some fears of black markets and commodity trading in addresses. Recovery of unused space is not the answer either, since route fragmentation has caused router tables to explode, bumping up against the capacity of the routers. Address recovery and reallocation only aggravates the routing table congestion problem while providing no significant relief from the address exhaustion problem. Routers are bursting at the seams.

This data on IPv4 and IPv6 BGP advertisements from January of 2007 through May of 2010 is derived from the data collected by the Asia Pacific Network Information Center, APNIC, and generated by the custom graph generator of the CIDR-Report project, **www.cider-report.org**. APNIC has data on IPv6 statistics from 2003 and data on IPv4 statistics from 1998 through this writing.

IPv6 expansion and deployment

The new generation Internet, IPv6, has been around for many years and has been enjoying a continued expansion not only in Europe and Asia but also in the US and elsewhere. Many years ago, the number of networks routable by IPv6 exceeded the number of routable IPv4 addresses but with a fraction of the routing table load. 2009 saw further deployment of IPv6 in government and defense. At current capacities, IPv6 has the capacity to handle the entire older IPv4 Internet several times over and is not bumping up against these limits.

The following two graphs—IPv4 on the left and IPv6 on the right—show the number of advertised routes in the core Internet routers through the Border Gateway Protocol (BGP). This data confirms that the number of routes for each protocol continues to expand.

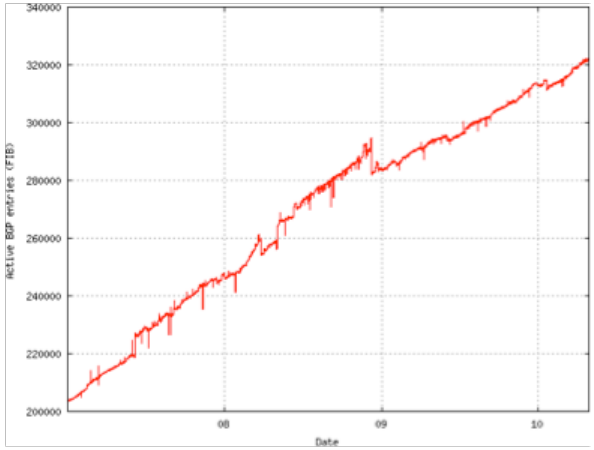


Figure 22: IPv4 BGP Advertisements January 2007-May 2010. Source: Asia Pacific Network INformation Center/CIDR-Report project

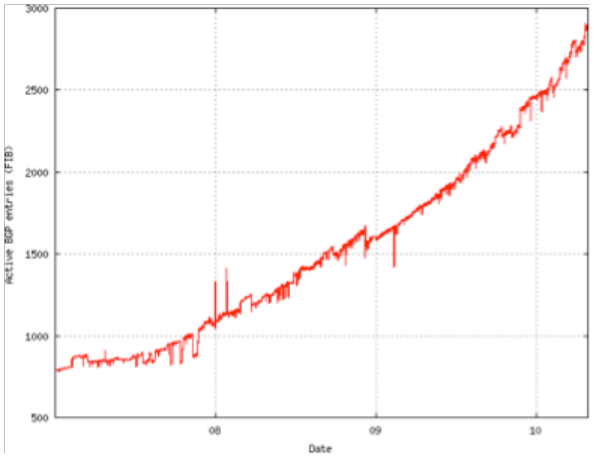


Figure 23: IPv6 BGP Advertisements January 2007-May 2010. Source: Asia Pacific Network INformation Center/CIDR-Report project

However, observe the labeling of the vertical axis. There are currently over 300,000 IPv4 routes advertised in the core Internet while somewhat less than 3,000 IPv6 routes are advertised. While requiring less than 1/100 the number of routes, these advertisements are routing roughly the same number of full /48 network addresses in IPv6 as there are single IPv4 host addresses. Each and every advertised IPv4 address could have an entire massive IPv6 network at the cost of less than 1/100 the number of routes in the routers. Due to the capacity of each IPv6 network, it's meaningless to compare the number of IPv4 addresses with even a single /48 IPv6 network. While the expansion of the number of IPv4 routes may appear to be slowing down slightly, the expansion of IPv6 routes and the number of advertised networks seems to be accelerating.

All modern operating systems support IPv6 and most networks already have IPv6 present on them, especially with Windows Vista, Windows 7, Mac OS/X, and Linux present. Unfortunately, people still ignore it and think it's something in the future. Without intending to, most networks have already deployed IPv6 by default. This is a trend that has accelerated with Vista and Windows 7 over the last year and should continue. Operators who are unaware of it or choose to ignore it are at risk as it deploys throughout their infrastructure.

The cable and broadband provider, Comcast, has been using IPv6 for years to manage devices internally, having run completely out of addresses in the 10.*.*.* private address space for managing their devices. They have now opened up a beta test program, offering IPv6 to their end users and customers. Comcast also has IPv6 Adoption Monitor" to track IPv6 deployments. You can learn more at the following link:
http://ipv6monitor.comcast.net/

Hurricane Electric, a popular ISP with connections in Europe, Asia, and Australia, offers an expanding array of IPv6 tools and facilities as well as a free tunnel broker service, **www.tunnelbroker.net**, offering free IPv6 connectivity. On their site is a "doomsday clock" that counts down to the exhaustion of the IPv4 address space and the number of days left (along with other numerous IPv4 and IPv6 statistics). They also have a free "certification" for individuals and organizations to train and test themselves in IPv6 knowledge and networking.

A number of prominent websites such as Google and YouTube are now fully IPv6 enabled. Google recently reported that the United States is fifth in the world for IPv6 deployment, largely as a result of Apple Macs and wireless access points which are already enabled for IPv6 and which automatically connect through one of the established automatic transition tunnels. Windows Vista and Windows 7 also automatically connect to the Teredo transition mechanism when native IPv6 is unavailable. While the percentages of client systems which prefer IPv6, when it's available, remain small, it is continuing to expand.

All of this points to a continuing expansion of IPv6 in the coming years and this trend is not going to slow down. Some time ago, IPv6 was referred to as the "Next Generation" IP protocol. It could be argued that IPv6 is now the "Current Generation" IP protocol while IPv4 is becoming the "Old Show."

Virtualization—consolidating into virtual spaces and what it means to our security

Virtualization technology is growing in importance. According to a recent IDC press release[1], 18.2 percent of all new servers shipped in the fourth quarter of 2009 were virtualized, representing a 20 percent increase over the 15.2 percent shipped in the fourth quarter of 2008. The size of the virtualization market in 2009 was US$15.2 billion. Growing interest in cloud computing will fuel further demand for virtualization solutions. Therefore, it is increasingly important to understand the security implications of virtualization technology. This section presents an analysis of vulnerability disclosures over the past decade for virtualization products provided by the following vendors:

- Citrix
- IBM
- Linux VServer
- LxCenter
- Microsoft
- Oracle
- Parallels
- RedHat
- VMware

Virtualization vulnerability disclosure trend

From 1999 through the end of 2009, 373 vulnerabilities affecting virtualization solutions were disclosed. The trend in the number of virtualization vulnerability disclosures is shown in Figure 24. These disclosures represent a small fraction of all disclosures, having exceeded the 1 percent level only in 2007 through 2009.

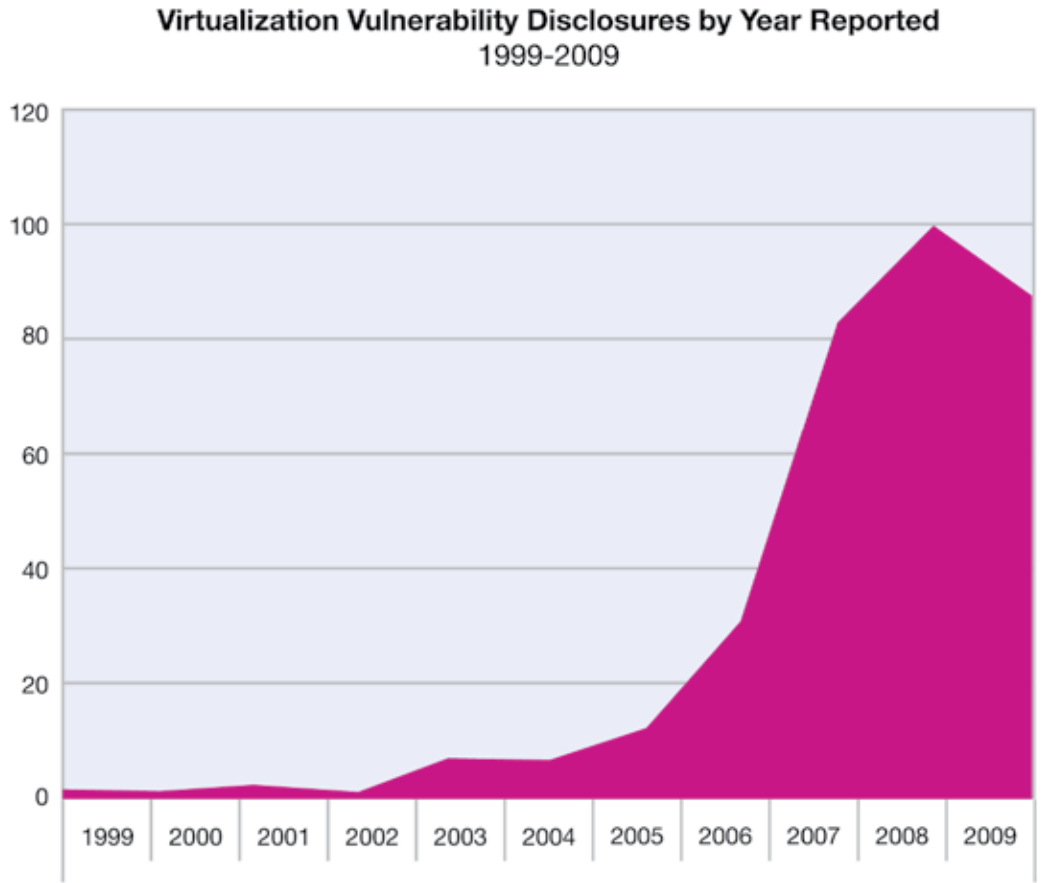


Figure 24: Virtualization vulnerability disclosures by year reported, 1999-2009

[1] http://www.idc.com/getdoc.jsp?containerId=prUS22316610

It is natural to expect that the number of vulnerability disclosures would have increased each year since virtualization products appeared on the market. While this was true from 2002 through 2008, the number of disclosures peaked in 2008 at 100, fell by 12 percent to 88 in 2009, and appears on track to fall slightly further in 2010 (39 virtualization vulnerabilities were disclosed in the first half of 2010). This trend in virtualization vulnerability disclosures suggests that virtualization vendors have been paying more attention to security since 2008 and/or security researchers have focused their efforts on easier targets.

Virtualization vulnerabilities by severity

As illustrated in Figure 25, high and medium severity vulnerabilities have made up over half of virtualization vulnerabilities in every year included in this analysis. High severity vulnerabilities have made up over one-third of all vulnerabilities in every year except 2006. These distinctions also hold true in the first half of 2010. Overall, 40 percent of reported vulnerabilities have high severity, 26 percent medium, and 34 percent low. Since high severity vulnerabilities tend to be easiest to exploit and provide full control over the attacked system, virtualization vulnerabilities represent a significant security threat. This is especially true considering that a number of these vulnerabilities negate the isolation normally provided by virtualization, making it possible to gain access to data outside the scope of an exploited virtual machine.

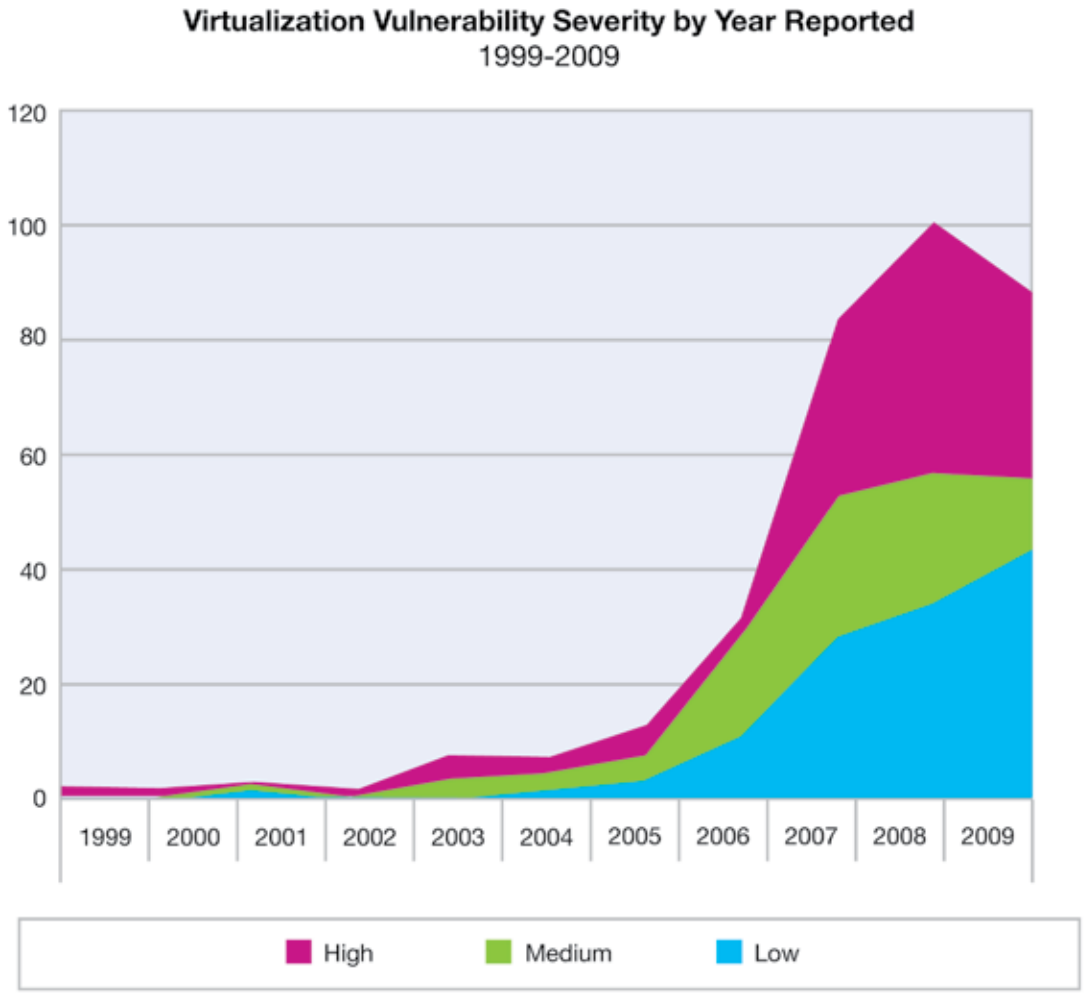


Figure 25: Virtualization vulnerability severity by year reported, 1999-2009

Virtualization vulnerabilities by location

It is important to understand the location of virtualization vulnerabilities (that is, where they occur in the code), since this affects how easily they can be remediated by vendors. Figure 26 compares the number of vulnerabilities in virtualization product vendor code against the number of vulnerabilities in third-party components used in virtualization products. In every year since 2005 (with the exception of 2007), the number of vulnerabilities in third party components has exceeded the number in vendor code. This distinction also almost held true in the first half of 2010, when there were 20 vulnerabilities in vendor code and 19 in third party components. This suggests that virtualization vendors need to be careful in choosing third party components, and should have mechanisms in place for quickly updating these components when vulnerabilities in them are reported.

These statistics break down differently for workstation and server products. Workstation products include those that run on top of a host operating system, and server products include those that run "on the bare metal" (that is, the hypervisor itself functions as an operating system). Workstation product vulnerabilities show a trend opposite that indicated in the graph—only 24 percent occur in third party components. Server product vulnerabilities exhibit this trend to an extreme degree, in that 70 percent of these vulnerabilities occur in third party components.

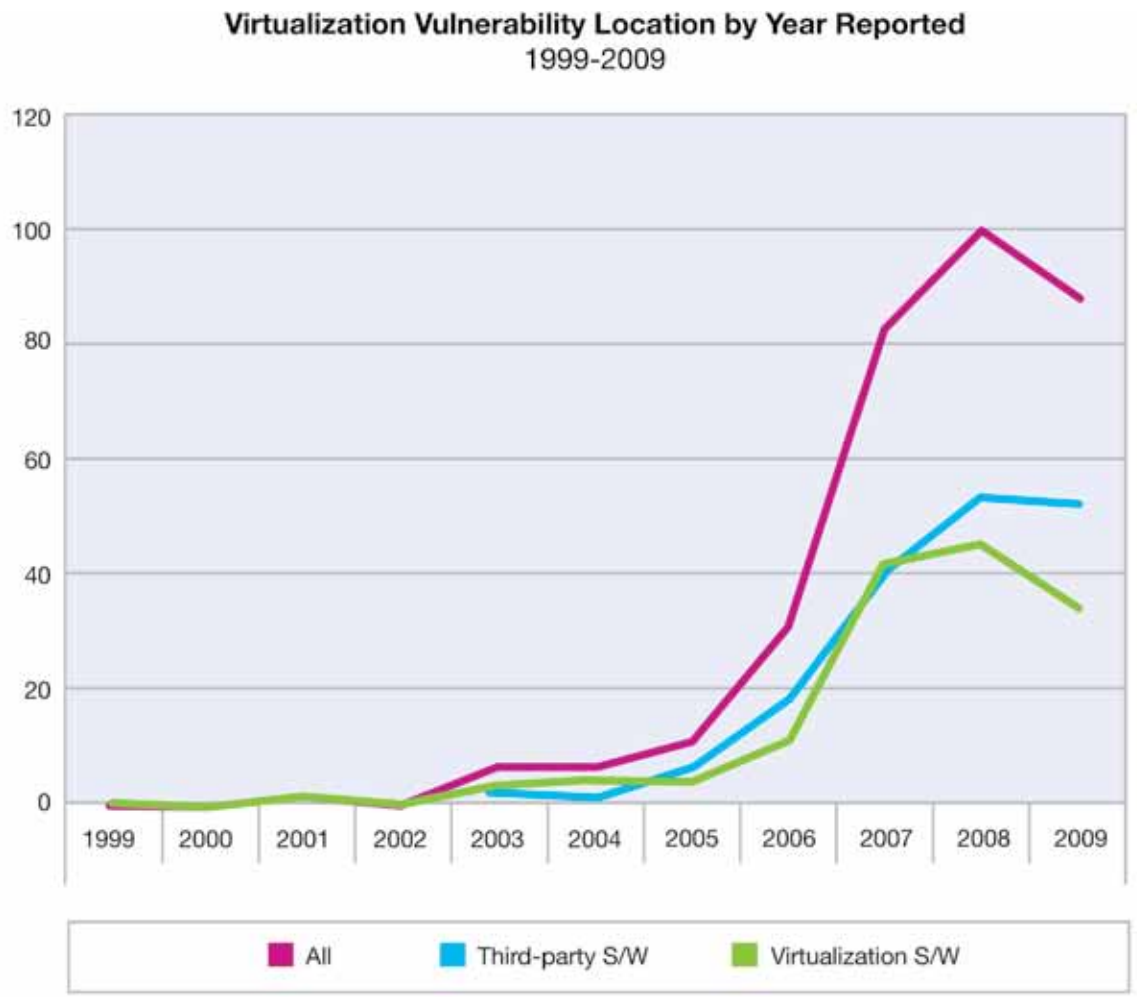


Figure 26: Virtualization vulnerability location by year reported, 1999-2009

Virtualization vulnerabilities by product type

Figure 27 shows the trend in workstation product vulnerabilities compared to server product vulnerabilities. As mentioned above, workstation products include those that run on top of a host operating system, and server products include those that run "on the bare metal" (that is, the hypervisor itself functions as an operating system). In every year since 2005, vulnerabilities in virtualization server products have overshadowed those in workstation products. This likely reflects the greater complexity of server products as well as a stronger focus on identifying server product vulnerabilities.

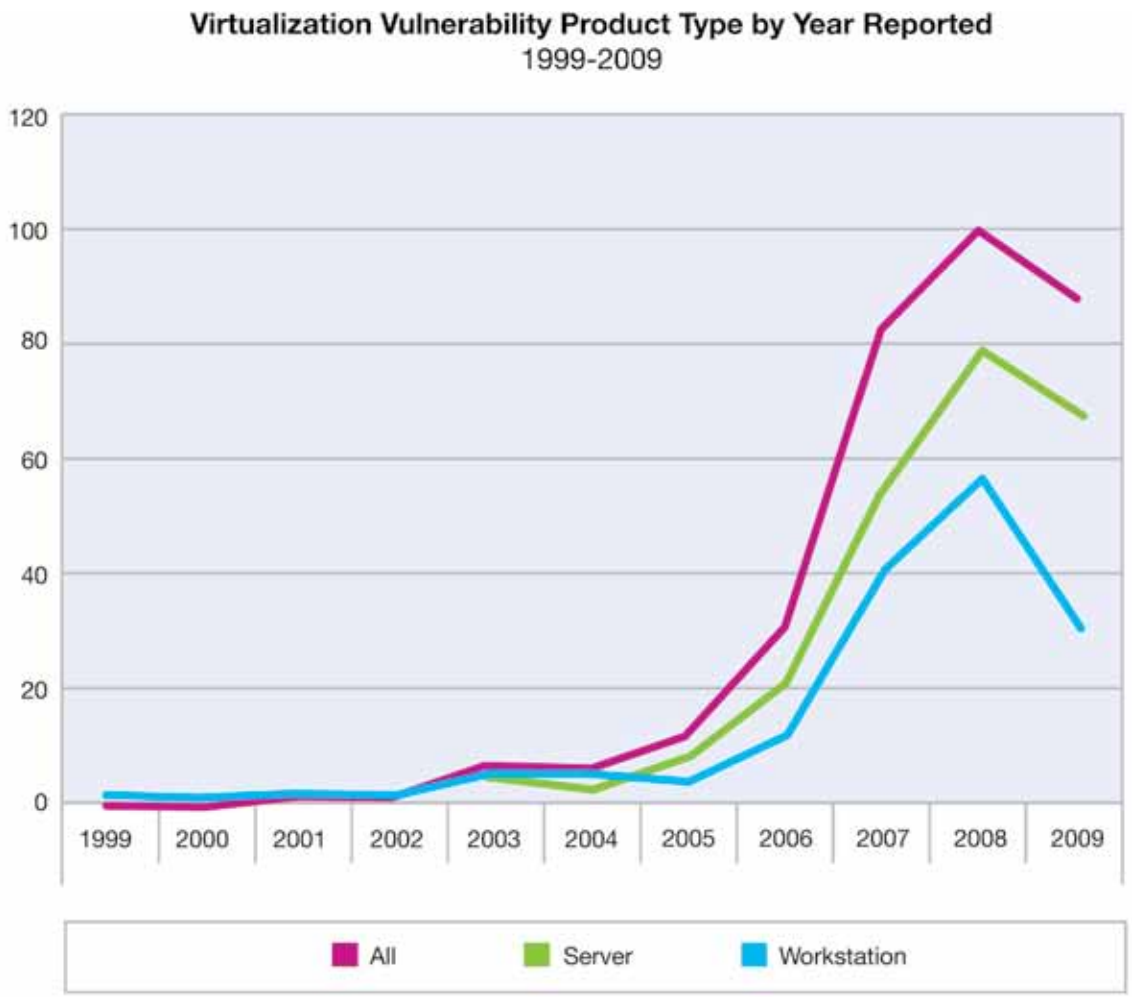


Figure 27: Virtualization vulnerability product type by year reported, 1999-2009

Virtualization vulnerabilities by vulnerability type

Respectively, Figures 28 and 29 show the distribution of vulnerabilities by vulnerability type for workstation and server products. This analysis includes only those vulnerabilities that exist in virtualization system code (vulnerabilities in third party components are excluded).

The defined vulnerability types, and the percentage each accounts for in workstation and server products, are given in Table 5.

Type	Description	Workstation Percentage	Server Percentage
Host	Vulnerabilities that affect the host operating system on which the virtualization system is installed without the involvement of any executing virtual machines.	30.8%	0%
Guest	Vulnerabilities that affect a guest virtual machine without affecting the hypervisor or host operating system.	26.3%	15.0%
Escape to host	Vulnerabilities that allow an attacker to "escape" from a guest virtual machine to affect the host operating system on which the virtualization system is running.	24.1%	0%
Web application	Vulnerabilities in Web applications (typically management applications) that affect the system on which the client browser is running.	9.8%	10%
Virtualization system	Vulnerabilities that affect the virtualization system itself, that is, the entire virtualized environment, but do not arise from guest virtual machines.	4.5%	37.5%
Escape to hypervisor	Vulnerabilities that allow an attacker to "escape" from a guest virtual machine to affect other virtual machines, or the hypervisor itself. In the case of workstation products, these vulnerabilities do not affect the host operating system.	3.8%	35.0%
Console	Vulnerabilities that affect custom management consoles.	0.8%	0%
Web server	Vulnerabilities that affect a Web server that implements a Web application used by the virtualization system.	0%	2.5%

Table 5: Virtualization vulnerabilities by vulnerability type for workstation and server products

Vulnerability type impacts

Host vulnerabilities, Web application vulnerabilities, Web server vulnerabilities, and console vulnerabilities are not unique to virtualization systems; they are analogous to similar vulnerabilities in traditional applications. The vulnerabilities that affect only remote components (Web application and console vulnerabilities) do not pose any greater risk than in traditional applications. Host vulnerabilities and Web server vulnerabilities pose server-side risks that are similar to those posed by traditional applications, but also hold the potential to affect multiple virtual machines running under the virtualization system. Guest machine vulnerabilities, escape-to-hypervisor vulnerabilities, escape-to-host vulnerabilities, and virtualization system vulnerabilities and are unique to virtualization systems and require additional analysis to understand the risks they pose.

Guest machine vulnerabilities

Guest machine vulnerabilities are analogous to host vulnerabilities in non-virtualized systems, because they affect only applications running on the affected guest machine. In this sense they do not pose a new type of risk—vulnerabilities in a system affect only that system.

Escape-to-host vulnerabilities

Escape-to-host vulnerabilities pose a new type of risk in the sense that a vulnerability in one system (a guest virtual machine) can affect the security of another system (the virtualization system's host) without propagating across a network. Vulnerability assessments run against the host operating system will fail to reveal all of the host's vulnerabilities. If an escape-to-host vulnerability exists, then the risk profile of the host includes additional risks associated with the virtual machines running on that host. This risk may vary over time as virtual machine images are started and stopped.

Escape-to-hypervisor vulnerability

Escape-to-hypervisor vulnerabilities, like escape-to-host vulnerabilities, involve the potential of one system (a guest virtual machine) to affect others without propagating across a network. In this case, the risks to virtual machines running under the same hypervisor depend on the vulnerabilities that exist in other virtual machines running under the same hypervisor.

Virtualization system vulnerabilities

Finally, virtualization system vulnerabilities pose a type of risk similar to that of host vulnerabilities—their potential impact extends beyond the virtualization system itself to the guest machines running under the virtualization system.

Workstation product vulnerabilities

Considering Figure 28, which presents workstation product vendor code vulnerabilities, we see that over half of these vulnerabilities fall into the first two categories, host and guest. These are somewhat traditional vulnerabilities in that they do not involve propagation of threats from virtual machines. What might prove surprising is that over 25 percent of workstation product vendor code vulnerabilities involve an escape from a virtual machine. In this product class, escape-to-host vulnerabilities are six times more common than escape-to-hypervisor vulnerabilities.

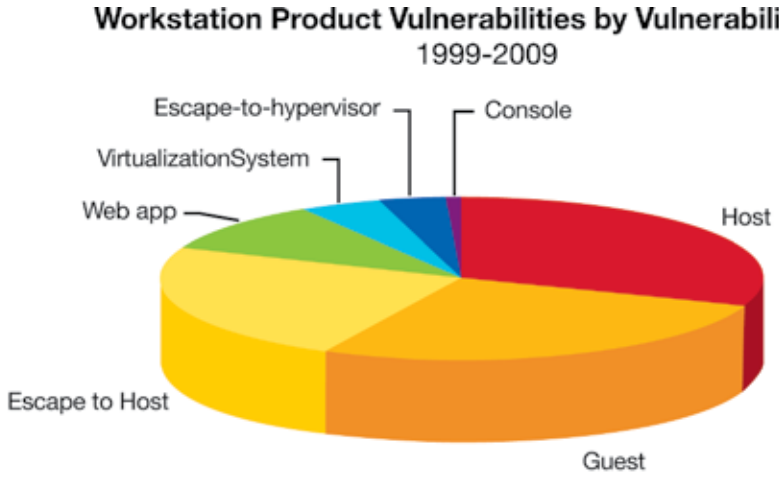


Figure 28: Workstation product vulnerability by vulnerability type, 1999-2009

Server product vulnerabilities

Considering Figure 29, we see that the largest class of vendor code vulnerabilities in server products is virtualization system vulnerabilities, which make up 38 percent. Escape-to-hypervisor vulnerabilities run a close second at 35 percent. Escape-to-hypervisor vulnerabilities make up over one-third of server product vendor code vulnerabilities. Server class escape-to-hypervisor vendor code vulnerabilities have affected products from Citrix, Parallels, RedHat, and VMware. Five of them are denial-of-service vulnerabilities, and one involves remote code execution.

The fact that server class escape-to-hypervisor vulnerabilities exist has implications for deployment of virtual servers. Within the market, there has been speculation that there are no escape-to-hypervisor vulnerabilities affecting server class systems, and therefore it is acceptable to run virtual servers with different security sensitivities on the same physical hardware. The results presented here show that escape-to-hypervisor vulnerabilities do exist for server class systems, calling into question whether virtual servers with different levels of security sensitivity should run on the same physical machine. This observation emphasizes the importance of insuring that virtual servers are not compromised, underscoring the importance of timely patch management for virtualization systems.

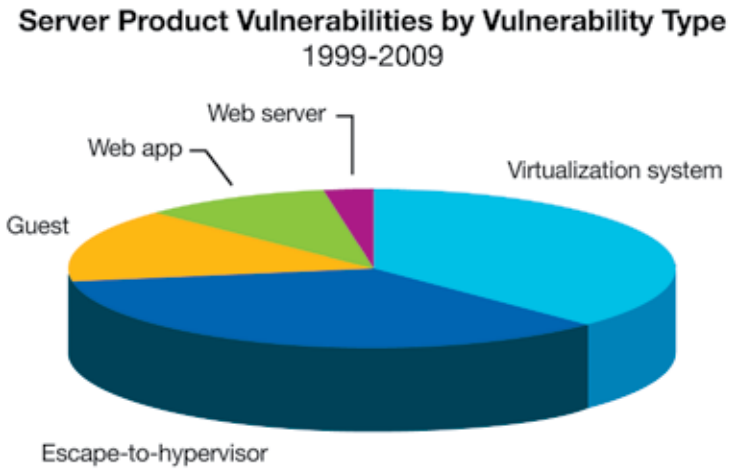


Figure 29: Server product vulnerabilities by vulnerability type, 1999-2009

Virtualization vulnerabilities by vendor

Figure 30 shows the contribution of virtualization vulnerability disclosures by each vendor included in this analysis. It is not surprising that the majority of vulnerabilities have been reported in VMware products, given VMware's position as market leader. VMware products accounted for over 80 percent of reported vulnerabilities, while the next nearest, RedHat and Citrix, accounted for about 7 percent and 6 percent respectively. All of the remaining vendors (including IBM, Microsoft, and Oracle/Sun) fared well, each accounting for only about 1 percent of reported vulnerabilities. Either these vendors have done an excellent job of addressing security in their products, or their offerings have not yet come under much scrutiny from vulnerability researchers.

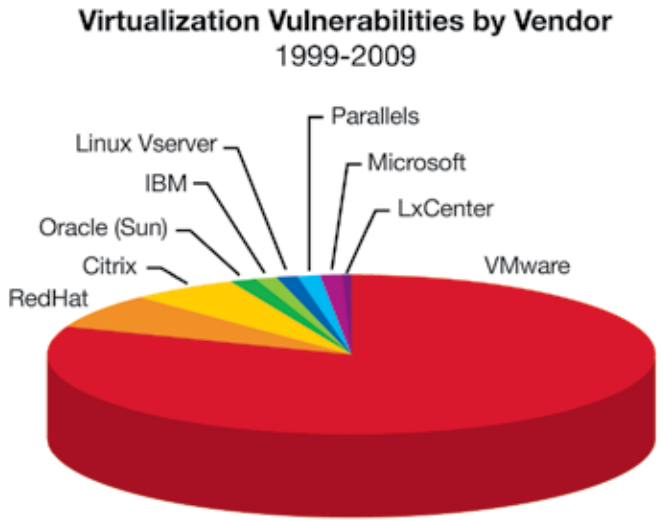


Figure 30: Virtualization vulnerabilities by vendor, 1999-2009

Exploit availability

The number of exploits known against a class of vulnerabilities provides one measure of how likely those vulnerabilities are to be exploited. Of the 373 virtualization vulnerabilities reported since 1999, 51 (14 percent) have known exploits. This compares to 25 percent of vulnerabilities in the entire X-Force database for which exploits are known. Therefore the incidence of exploit availability for virtualization vulnerabilities is about half that of vulnerabilities at large. This reflects an inherently greater difficulty in exploiting virtualization vulnerabilities and/or a lesser focus on virtualization products by exploit developers.

One class of vulnerabilities of particular interest is escape-to-hypervisor vulnerabilities in server products, since these have extremely high risk. Of the 28 vulnerabilities of this type, only 2 have known exploits. While this represents a very small fraction, the fact that exploits exist for this class of vulnerabilities is cause for concern.

The emerging cloud: adoption of cloud services for the future

Cloud computing represents the latest disruptive technology to hit the market, largely driven by the cost benefits and efficiencies that organizations can gain. As with all emerging technology, the excitement introduced needs to be managed to ensure that the technology does not introduce new risks and implications to the organization. Not surprisingly, many organizations are delaying migration to the cloud and merely testing the waters of cloud computing.

Recent studies show that organizations looking to adopt the technology believe security represents the greatest impediment to the cloud, followed closely by availability. In reality, few enterprises have fully adopted the cloud paradigm, and most businesses are currently transitioning only those elements that have little risk to the business. Inversely, we see a number of small and medium businesses benefiting from the cloud, with the expectation that they can increase their security as a by-product of the capabilities provided by the hosting organization.

The hesitancy of widespread adoption of cloud based technologies is deeply rooted in the complexity and variability of cloud offerings and their capabilities. Organizations often begin their assessment of the cloud by looking at individual vendors and capabilities. However, at IBM we assert that organizations should initiate the adoption of cloud through the consideration of the workloads intended for the cloud.

By assessing a cloud based on its workload, organizations can better understand the factors necessary for selecting a suitable cloud deployment scenario. For example, organizations looking at workloads that contain health care data can define their security and audit needs and outline any requirements of joint commitment regarding regulatory constraints. This type of consideration offers other benefits for organizations such as an increased understanding of the data within their organization and its relevance to the business.

After categorizing the data in terms of security and regulatory requirements, organizations can leverage this information to determine the attributes essential for protecting their data in the cloud, and can formulate criteria for evaluating various deployment providers. An example of an attribute that may apply would be where an organization has specific eDiscovery requirements or obligations whereby certain data must be preserved and made available for legal reasons.

Additional concerns for those deploying in the public cloud are factors such as the financial stability of the hosting organization and the hosting organization's deployment policies. For example, a customer might want to avoid any vendor which clusters customers in groups, as legal issues affecting one tenant could impact other co-tenants.

In closing, its important that organizations take a strategic approach to adopting cloud based services. This means developing a strong understanding of opportunities and requirements before the search for a vendor begins. By doing your homework up front, you can be better prepared to identify the right business partners.

Section II
Overview

The IBM X-Force® research and development team discovers, analyzes, monitors, and records a broad range of computer security threats and vulnerabilities. According to X-Force observations, some new trends have surfaced in the first half of 2010. We hope the information presented in this report regarding these trends provides a useful foundation for planning your information security efforts for the rest of 2010 and beyond.

2010 Mid-year highlights

Vulnerabilities

- The number of new vulnerability disclosures in the first half of the year is at the highest level ever recorded. This is in stark contrast to the 2009 mid-year report when new vulnerability disclosures were at the lowest level in the previous four years. Web application vulnerabilities—particularly cross-site scripting and SQL injection—continue to dominate the threat landscape.
- Apple is maintaining the top spot of vendor with the most vulnerability disclosures accounting for a full four percent of all disclosures. After three years of holding the number one position of vendor with the most vulnerability disclosures, Microsoft has dropped to number two. Adobe is in third place, due to the noteworthy increase in reported PDF and Flash-based vulnerability disclosures.
- As for operating systems, Linux took the number one position in the first half of this year for new operating system disclosures followed by Apple in second place. If you consider only the critical and high operating system disclosures, Microsoft dwarfed all the other players with 73 percent.

Exploitation

- Web applications continue to account for 55 percent of all vulnerability disclosures.
- PDF exploitation is prevalent in 2010 with attackers using a mixed bag of tricks of spam, phishing, and obfuscation all disguised to confuse the end-user.
- Internet Explorer takes the early lead in vulnerability disclosures and in attackers actively seeking methods to exploit those new vulnerabilities.

Vulnerabilities

First half of 2010 vulnerability disclosure count

Earlier in Section I of this report we discussed the incredible number of vulnerabilities already disclosed in the first half of 2010. We expect that 2010 will set records for high disclosure numbers.

In the Web section of the report, we discuss the vulnerabilities affecting Web applications which represent half of all reported disclosures. Directly behind those Web applications are customer developed applications, Web browsers, and PDFs—all of which continue to demonstrate record reporting in this first half of 2010.

To avoid any ambiguity regarding the characterization of vulnerabilities, this report uses the following IBM Security Services definition.

Vulnerability is defined as a set of conditions that leads or may lead to an implicit or explicit failure of the confidentiality, integrity, or availability of an information system.

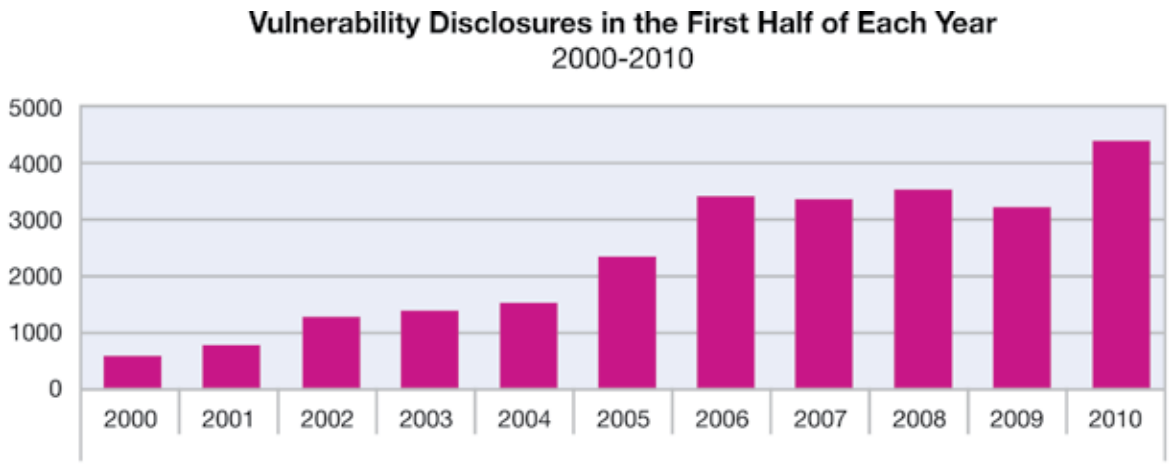


Figure 31: Vulnerability disclosures in the first half of each year, 2000-2010

Vulnerability disclosures by severity

The Common Vulnerability Scoring System (CVSS) is the industry standard for rating vulnerability severity and risk based on formulas as well as both base and temporal metrics. Base metrics are characteristics that generally do not change over time such as access vector, complexity, authentication, and the impact bias. Temporal metrics are the characteristics of a particular vulnerability that can and often do change over time, and include exploitability, remediation level, and report confidence.

Vulnerabilities identified as Critical by CVSS metrics are vulnerabilities that are installed by default, are network-routable, do not require authentication to access, and that allow an attacker to gain system or root-level access.

Table 6 represents the severity level associated with both the base and temporal CVSS scores.

CVSS Score	Severity Level
10	Critical
7.0-9.9	High
4.0-6.9	Medium
0.0-3.9	Low

Table 6: CVSS Score and Corresponding Severity Level

For more information about CVSS, including a complete explanation of CVSS and its metrics, see the First.org website at **http://www.first.org/cvss/**

CVSS base scores

As Figure 32 indicates, Critical vulnerabilities maintained their one-percent position, similar to the percentages seen in both 2008 and 2009.

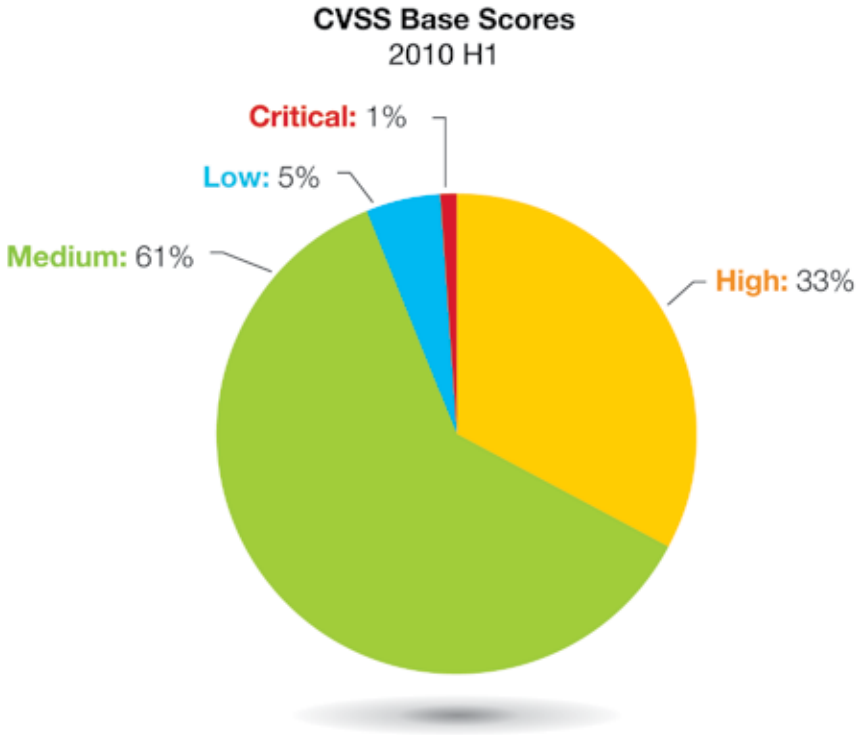


Figure 32: CVSS base scores, 2010 H1

Relative percentages are fairly consistent with 2009 data. There is a small drop in low and medium vulnerabilities with a corresponding slight increase in high vulnerabilities. Medium vulnerabilities contain the two most common vulnerability disclosures: SQL injection and cross-site scripting. High vulnerabilities increased to 33 percent compared with 30 percent in the first half of 2009 and 36 percent in 2008 as shown in Figure 33.

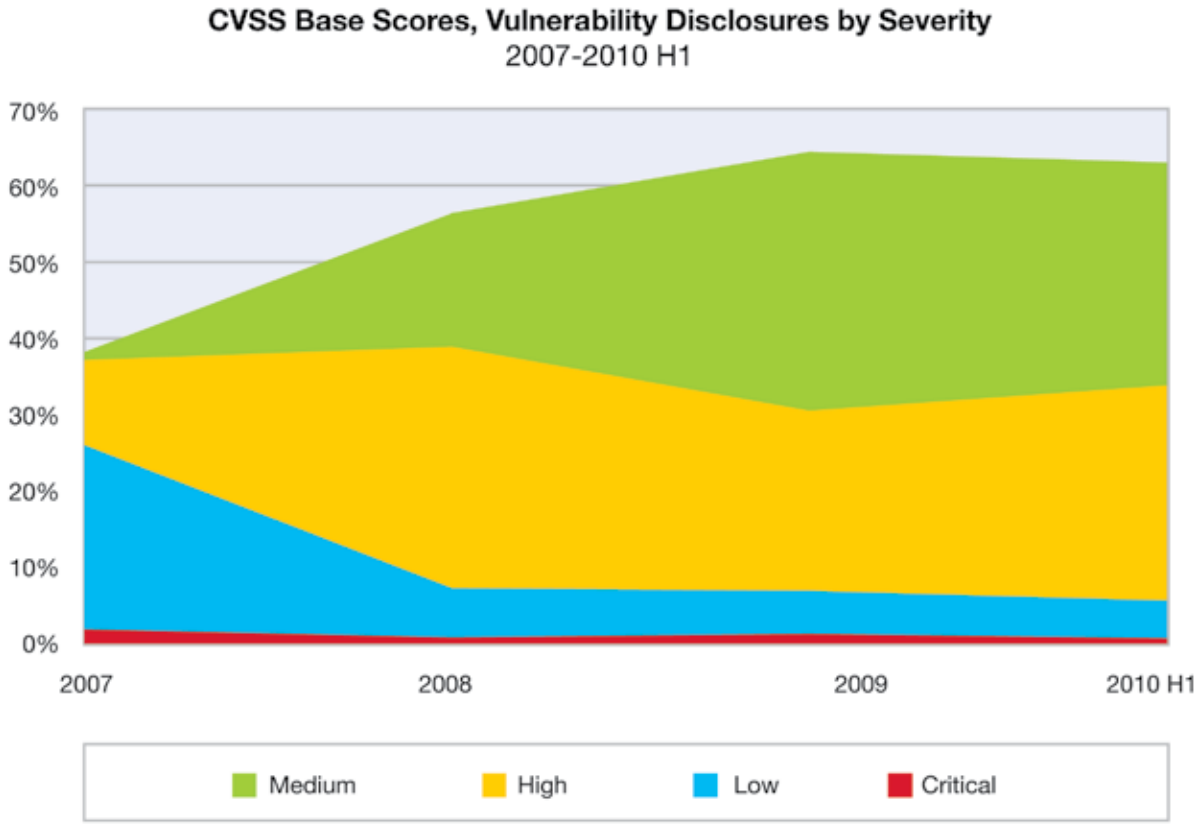


Figure 33: CVSS base scores, vulnerability disclosures by severity, 2007-2010 H1

Vendors with the most vulnerability disclosures

Vulnerability disclosures for the top ten vendors in the first half of 2010 accounted for a fifth of all disclosed vulnerabilities, down slightly from 2009 (23 percent), and up fractionally from 2008 (19 percent) and 2007 (18 percent).

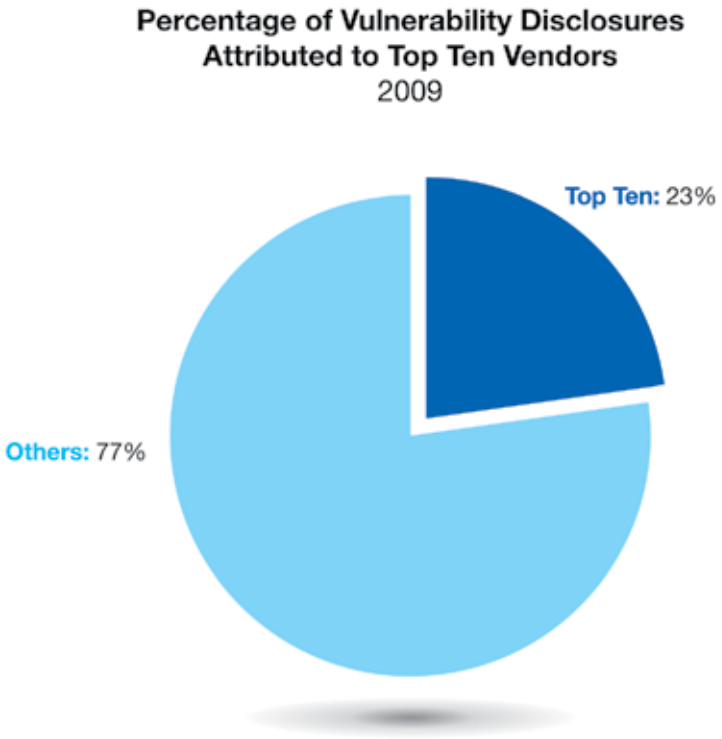


Figure 34: Percentage of vulnerability disclosures attributed to top ten vendors, 2009

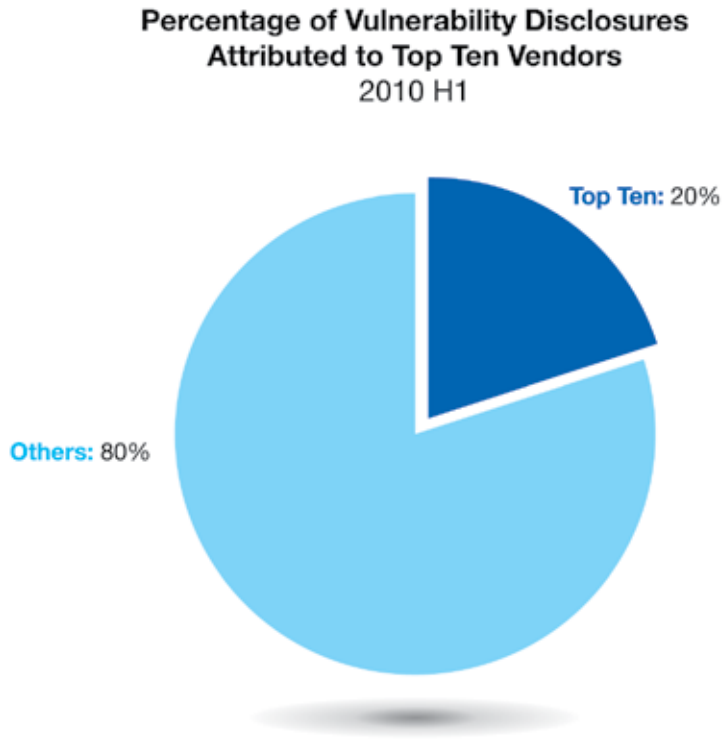


Figure 35: Percentage of vulnerability disclosures attributed to top ten vendors, 2010 H1

Changes in the top vendor list

The X-Force database team uses an industry standard called CPE (Common Platform Enumeration) to assign vulnerabilities to vendors and vendor products. See **http://cpe.mitre.org/** for more information.

Table 7 reveals the top ten vendors and their percentages of vulnerabilities in the first half of 2010 compared to 2009. Note that these statistics do not balance vulnerability disclosures with market share, number of products, or the lines of code that each vendor produces. Generally speaking, mass-produced and highly distributed or accessible software is likely to have more vulnerability disclosures.

Some observations:

- Apple maintained the top position for a second year in a row with a full four percent of all vulnerability disclosures.
- Sun dropped to the bottom of the list from second place last year—a significant change over 2009. Although Oracle acquired Sun in April 2009, their product lines have remained distinct in our database, so we continue to list them separately.
- Microsoft moved from third to second place after holding the top vendor spot in 2006-2008.

2010 H1		
Rank	**Vendor**	**Disclosure Frequency**
1.	Apple	4.0%
2.	Microsoft	3.4%
3.	Adobe	2.4%
4.	Cisco	1.9%
5.	Oracle	1.7%
6.	Google	1.6%
7.	IBM	1.5%
8.	Mozilla	1.4%
9.	Linux	1.4%
10.	Sun	1.1%

2009 (Full Year)		
Rank	**Vendor**	**Disclosure Frequency**
1.	Apple	3.8%
2.	Sun	3.3%
3.	Microsoft	3.2%
4.	IBM	2.7%
5.	Oracle	2.2%
6.	Mozilla	2.0%
7.	Linux	1.7%
8.	Cisco	1.5%
9.	Adobe	1.4%
10.	HP	1.2%

Table 7: Vendors with the most vulnerability disclosures

- Adobe moved up to third place from ninth, probably due to the significant increase in reported PDF and Flash-based vulnerabilities during the first half of 2010.
- HP dropped off the list while Google joined it at sixth place.

Availability of vulnerability fixes and patches

In Section I, we discussed the availability of vulnerability and patch rates. We demonstrated that major vendors are doing a solid job of addressing and fixing known vulnerabilities and we listed the best and worst patchers among the major vendors. Next, we address remotely exploitable vulnerabilities.

Remotely exploitable vulnerabilities

The most significant vulnerabilities are those that can be exploited remotely, because they do not require physical access to a vulnerable system. Remote vulnerabilities can be exploited over the network or Internet, while local vulnerabilities need direct system access. Vulnerabilities falling into both remote and local categories are those that can be exploited by both vectors.

In the past four and a half years, remotely exploitable vulnerabilities have grown from 85 percent to 94 percent of all vulnerability disclosures. 2009 remote vulnerabilities were at 92 percent and this has crept up to 94 percent in the first half of 2010. Figure 36 shows the steady growth in remotely exploitable vulnerabilities year over year for the last decade.

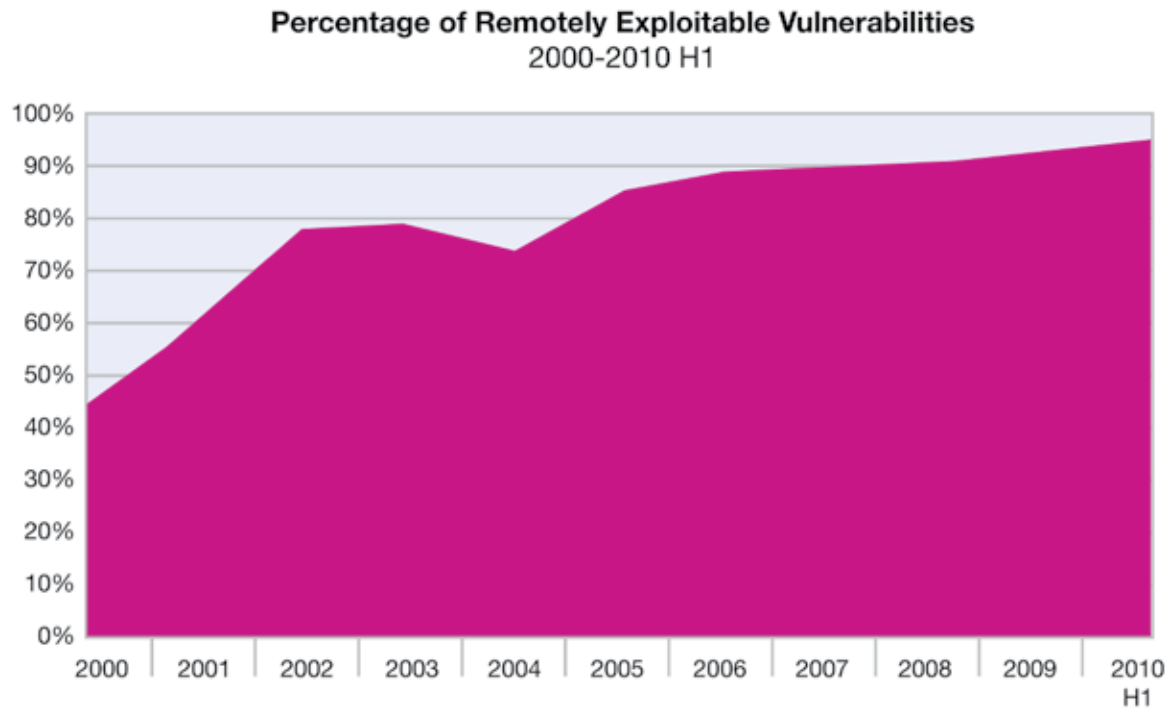


Figure 36: Percentage of remotely exploitable vulnerabilities, 2000-2010 H1

Exploitation consequences

IBM X-Force categorizes vulnerabilities by the consequence of exploitation. This consequence is essentially the benefit that exploiting the vulnerability provides to the attacker. Table 8 describes each consequence.

Consequence	Definition
Bypass Security	Circumvent security restrictions such as a firewall or proxy, IDS system, or virus scanner
Data Manipulation	Manipulate data used or stored by the host associated with the service or application
Denial of Service	Crash or disrupt a service or system to take down a network
File Manipulation	Create, delete, read, modify, or overwrite files
Gain Access	Obtain local and remote access. This also includes vulnerabilities by which an attacker can execute code or commands, because this usually allows the attacker to gain access to the system.
Gain Privileges	Privileges can be gained on the local system only
Obtain Information	Obtain information such as file and path names, source code, passwords, or server configuration details
Other	Anything not covered by the other categories

Table 8: Definitions for vulnerability consequences

The most prevalent consequence of vulnerability exploitation continues to be Gain Access, where it accounts for 52 percent of all vulnerability consequences. After a dip in 2008, gaining access is back above the 50 percent mark, where it was in 2006 and 2007. Gaining access to a system provides an attacker complete control over the affected system, which would allow him or her to steal data, manipulate the system, or launch other attacks from that system.

After peaking at 22 percent in 2008, vulnerabilities that allow an attacker to manipulate data are at 21 percent reflecting significant SQL injection activity.

Percentage-wise, most other attack vectors remain similar to previous years.

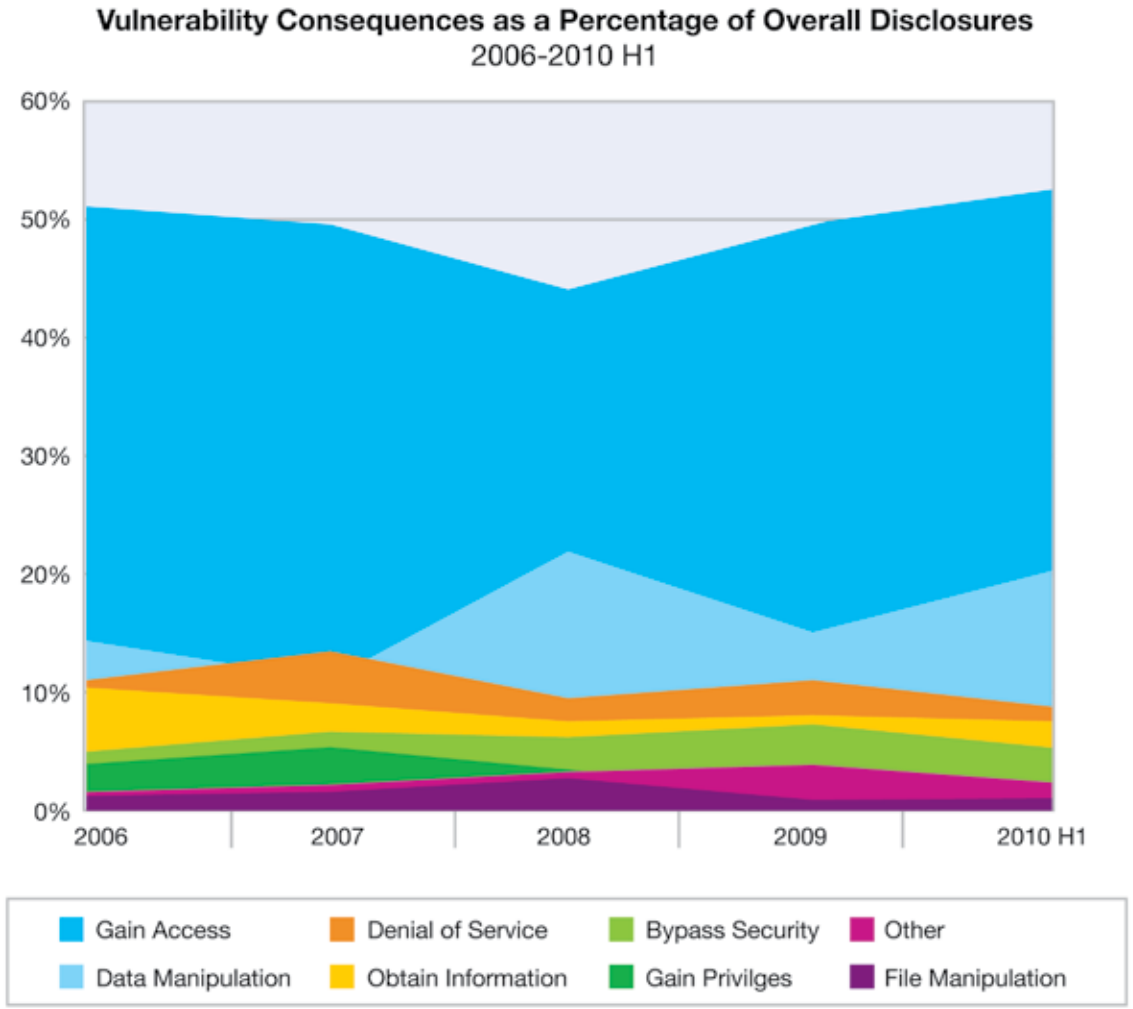


Figure 37: Vulnerability consequences as a percentage of overall disclosures, 2006-2010 H1

Operating systems with the most vulnerability disclosures

The following operating system analysis counts unique vulnerabilities reported for a single genre of operating system. For example, this analysis compares all vulnerabilities reported for Microsoft operating systems to all of the vulnerabilities reported for Apple operating systems in the same time period. If a certain vulnerability applies to multiple versions of operating systems in that genre, it is only counted once. For example, if a certain CVE applies to both Apple Mac OS X and Apple Mac OS X Server, it is only counted once for the Apple genre.

All operating systems vulnerabilities

For the first half of 2010, Linux had the largest percentage of operating system vulnerability disclosures followed closely by Apple in second place. Microsoft experienced a healthy spike over 2009 and moved into third place. Sun Solaris moved to fourth place with a significant drop in vulnerability disclosures. BSD maintained its number five slot, and IBM AIX, which was fifth in 2008, remains off the list for a second year in a row.

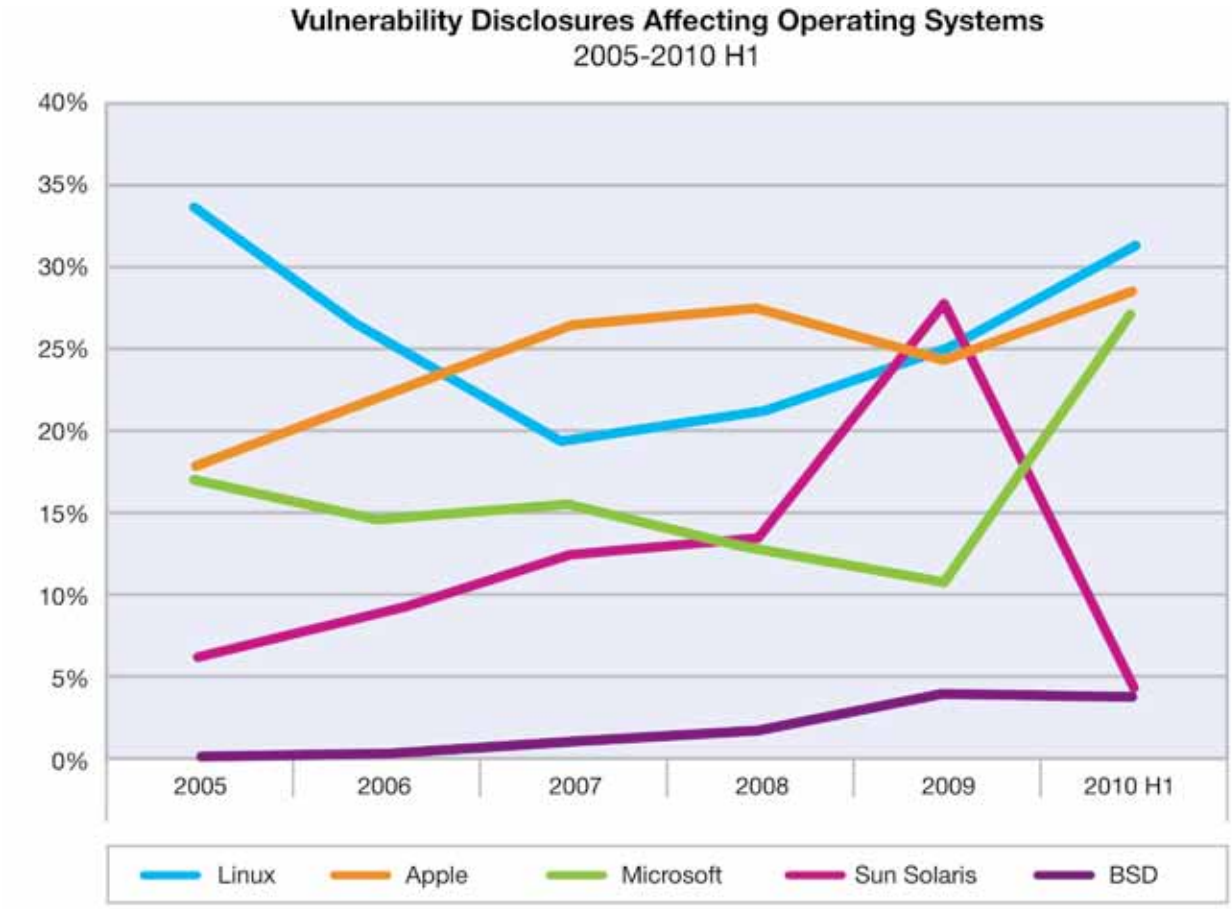


Figure 38: Vulnerability disclosures affecting operating systems, 2005-2010 H1

Critical and high operating system vulnerabilities

Focusing on critical and high vulnerabilities is another way to look at operating system vulnerabilities. From a protection standpoint, these high-severity vulnerabilities are typically the most worrisome since they often lead to complete remote compromise, the prized possession of attackers. When you filter out the mediums and lows, Microsoft operating systems take first place in 2008, in 2009, and in the first half of 2010. Linux is currently in second place with Apple coming in third place. HP-UX comes in fourth place and Sun Solaris trails closely in fifth place. IBM AIX dropped off the list after being in fifth place in the first half of 2009.

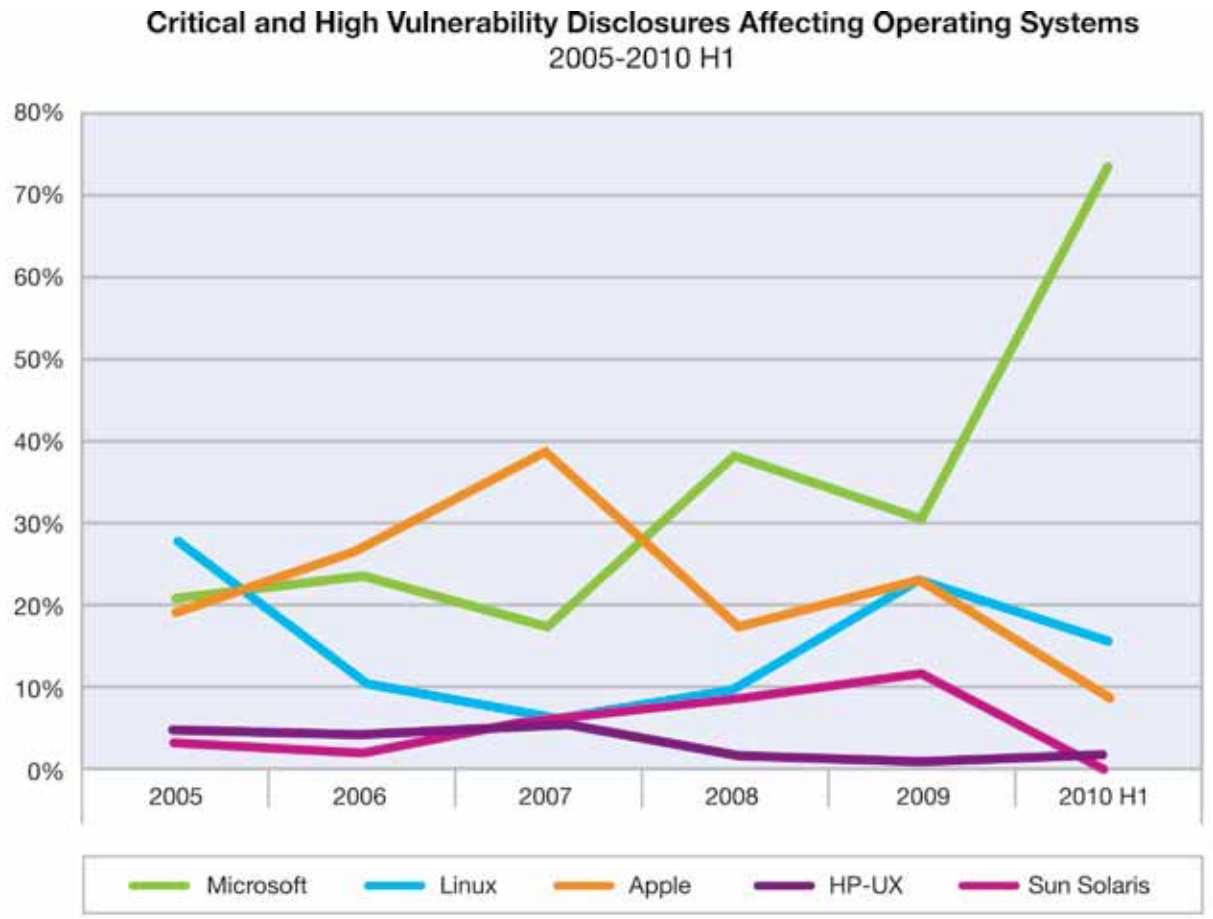


Figure 39: Critical and high vulnerability disclosures affecting operating systems, 2005-2010 H1

The top five operating systems listed in Table 9 account for 98 percent of all critical and high operating system vulnerability disclosures in the first half of 2010 compared to 93 percent for the first half of 2009. The top five operating systems account for 95 percent of all operating system vulnerability disclosures in the first half.

Operating System	Percentage of Critical and High	Percentage of all OS Vulnerabilities
Microsoft	73%	27%
Apple	9%	29%
Linux	16%	31%
HP-UX	2%	1%
Sun Solaris	0%	4%
BSD	0%	4%
IBM AIX	0%	2%
Others	2%	4%

Table 9: Operating systems with the most critical and high vulnerability disclosures, 2010 H1.

Why don't we use CPE to count operating systems?

Looking back to our 2008 report, X-Force presented an analysis of operating systems with the most vulnerabilities. These vulnerabilities were counted according to how each vendor reports their platforms through the Common Platform Enumeration (or CPE). There are slight differences in how some vendors classify their platforms. For example, Linux vulnerabilities that may be reported for the platform (Linux kernel) may also affect other Linux versions even though they may not be officially reported for that platform as it is reported in CPE. Other differences include the way that vendors classify a platform. Apple, for example, combines all versions of their Apple Mac OS X software into a single "platform" and only differentiates between the server and desktop versions of the software. Microsoft calls each of its major operating systems "platforms" even though some of these platforms may be considered by other individuals to be "versions" of Windows.

So, instead of counting vulnerabilities according to the named "platforms" in CPE, this report merges similar platforms together (all Windows, all Apple) and only counts a single vulnerability once, even if it affects multiple version of a particular genre of operating system.

Keeping operating system vulnerabilities in perspective

Operating system vulnerabilities always generate serious concern. But it is the diverse array of applications that run on operating systems that are the real problem, as many core statistics in this report make clear. Vulnerability disclosures for operating systems represent approximately 11 percent of all disclosed vulnerabilities for the first half of 2010. For many years, organizations have put patch operations in place to ensure that operating systems are patched and protected as soon as possible. So, although the operating system is ubiquitous software, these factors make them much more difficult to successfully attack. Other components, such as Web applications, Web browsers, and malicious documents including PDFs have pushed operating systems aside as the most worrisome threat vector.

Web application threats and vulnerabilities

Web application vulnerabilities continue to be the most prevalent type of vulnerability affecting servers today. As a percentage, Web application vulnerabilities have moved up past the 55 percent mark, accounting for more than half of all vulnerability disclosures in the first half of 2010.

The number of Web application vulnerabilities continues to climb at a moderately steady rate of 3,000 to 4,000 disclosures per year. These figures do not include custom-developed Web applications or customized versions of these standard packages, which also introduce vulnerabilities.

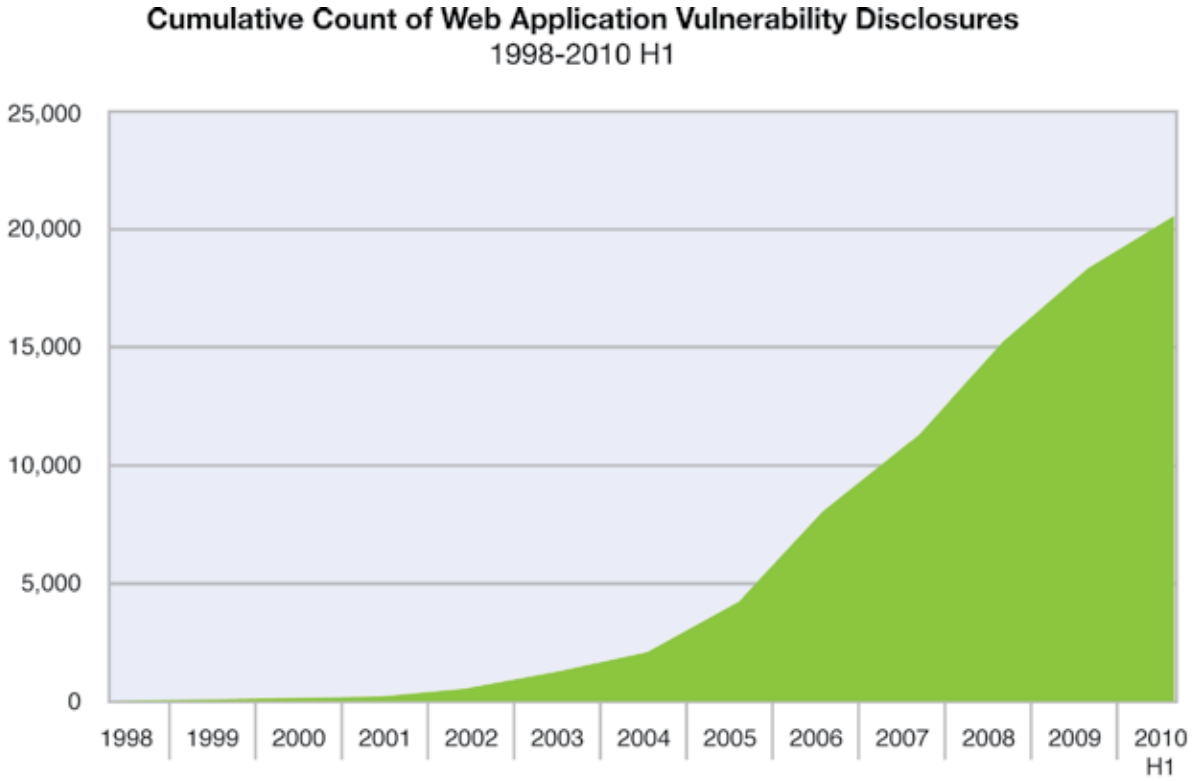


Figure 40: Cumulative count of Web application vulnerability disclosures, 1998-2010 H1

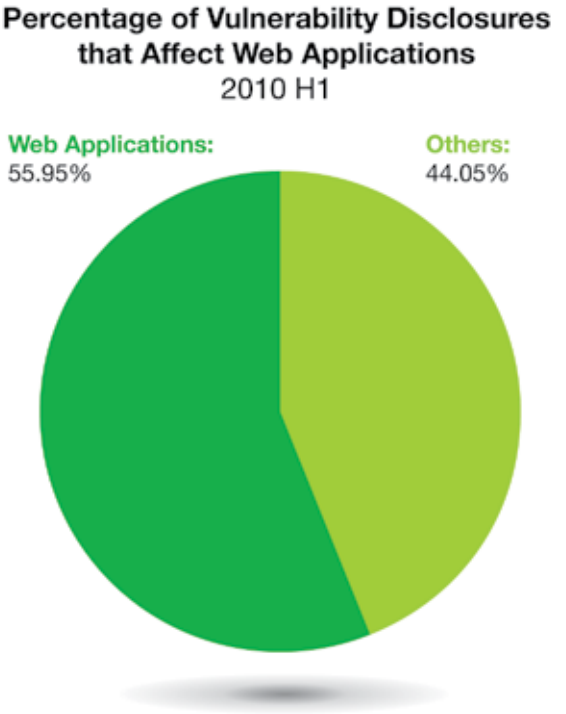


Figure 41: Percentage of vulnerability disclosures that affect Web applications, 2010 H1

Therefore, these vulnerabilities may only represent the tip of the iceberg of the total number of application vulnerabilities that exist on the Internet.

Web application vulnerability disclosures by attack categories

Cross-site scripting (XSS) and SQL Injection vulnerabilities are the predominate types of security vulnerabilities affecting Web applications in the first half of 2010.

Figure 42 illustrates the relative dominance of cross-site scripting, SQL injection, file include, and other vulnerability disclosures over time while Table 10 describes each category including the impact they can have on organizations and the customers they serve.

The previous X-Force trend report, published at the end of 2009 showed a significant decline in the disclosure of SQL injection vulnerabilities year over year. At the time we took this as a sign of progress. SQL Injection vulnerabilities have been the target of a great deal of exploitation activity on the Internet over the past few years, and a decline in disclosure might indicate that these vulnerabilities are getting harder to find—that some of the low hanging fruit has been plucked. Unfortunately, it appears that the volume of SQL injection disclosure is back up during the first half of 2010. Clearly we have yet to turn the corner on Web application vulnerabilities.

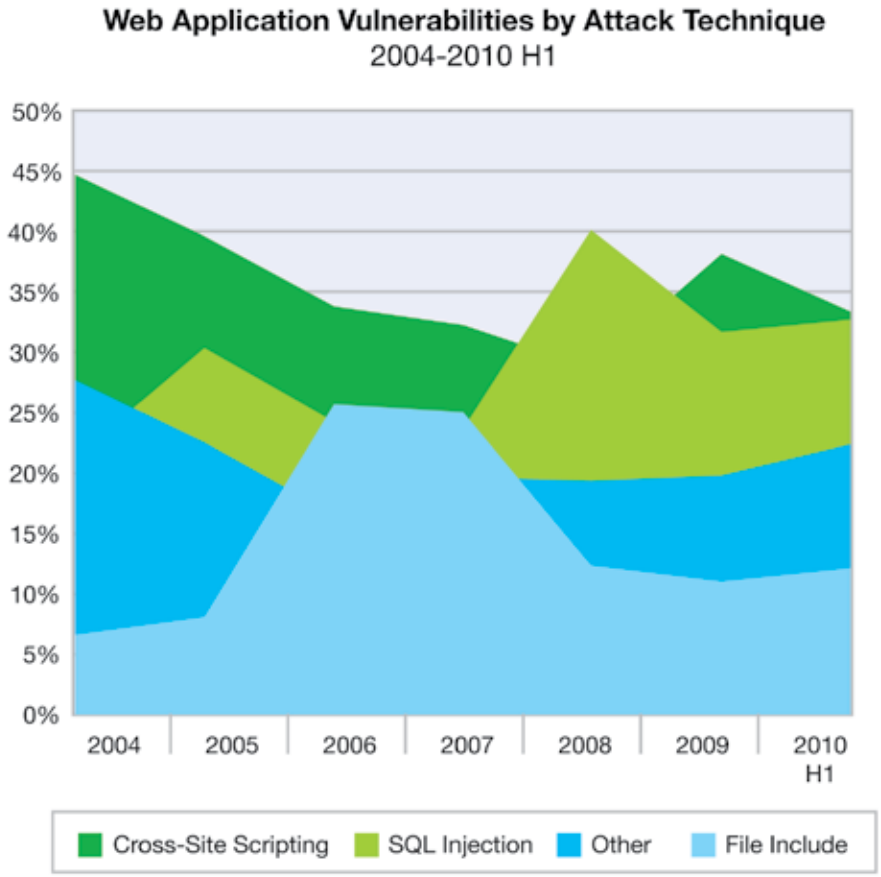


Figure 42: Web application vulnerabilities by attack technique, 2004-2010 H1

Cross-site scripting attacks on Web applications

The **Web Application Vulnerability Disclosures by Attack Categories** section notes that Cross-site scripting (XSS) vulnerability disclosures were quite prevalent in the first half of 2010. Real-world MSS data reveals that this type of Web Application vulnerability is also a favorite method of exploitation amongst attackers.

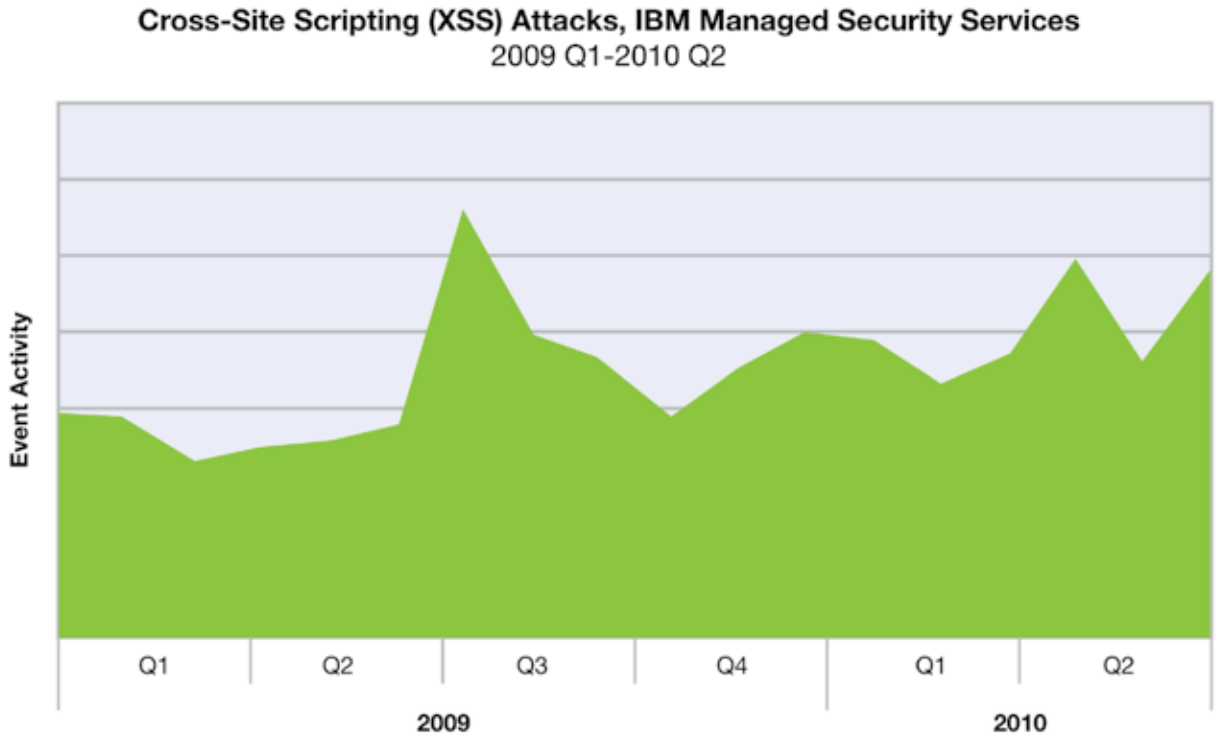


Figure 43: Cross-Site Scripting (XSS) attacks, IBM Managed Security Services, 2009 Q1-2010 Q2

Attack Technique	Description
Cross-site Scripting	Cross-site scripting vulnerabilities occur when Web applications do not properly validate user input from form fields, the syntax of URLs, etc. These vulnerabilities allow attackers to embed their own script into a page the user is visiting, manipulating the behavior or appearance of the page. These page changes can be used to steal sensitive information, manipulate the Web application in a malicious way, or embed more content on the page that exploits other vulnerabilities. The attacker first has to create a specially-crafted Web link, and then entice the victim into clicking it (through spam, user forums, etc.). The user is more likely to be tricked into clicking the link, because the domain name of the URL is a trusted or familiar company. The attack attempt may appear to the user to come from the trusted organization itself, and not the attacker that compromised the organization's vulnerability.
SQL Injection	SQL Injection vulnerabilities are also related to improper validation of user input, and they occur when this input—from a form field for example—is allowed to dynamically include SQL statements that are then executed by a database. Access to a back-end database may allow attackers to read, delete, and modify sensitive information, and, in some cases, execute arbitrary code. In addition to exposing confidential customer information (such as credit card data), SQL Injection vulnerabilities can also allow attackers to embed other attacks inside the database that can then be used against visitors to the website.
File Include	File Include vulnerabilities (typically found in PHP applications) occur when the application retrieves code from a remote source to be executed in the local application. Often, the remote source is not validated for authenticity, which allows an attacker to use the Web application to remotely execute malicious code.
Other	This category includes some denial-of-service attacks and miscellaneous techniques such as directory traversal and others that allow attackers to view or obtain unauthorized information, or change files, directories, user information or other components of Web applications.

Table 10: Description of the most prevalent categories of Web application vulnerabilities

OWASP Top 10

The increasing attacks targeted at Web applications, services and data are driving organizations to address security enforcement across the enterprise. These attacks include cross site scripting (XSS), SQL injection attacks, denial-of-service attacks, and miscellaneous techniques such as directory traversal and others. These types of attacks allow attackers to view or obtain unauthorized information, or to change files, directories, user information, and other components of Web applications.

The OWASP Top 10 (Open Web Application Security Project) provides an awareness document for Web application security and represents a broad consensus surrounding the most critical Web application security flaws. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Vulnerabilities such as broken authentication and session management allow attackers to compromise passwords, keys, session tokens, or exploit other implementation flaws to assume user's identities. Failure to restrict URL access, security misconfiguration, and un-validated redirects and forwards, expose business-sensitive data and information to unauthorized users. We recommend that organizations assess the risks and vulnerabilities of externalized applications and services and implement appropriate security controls to manage user identity and access within and across the enterprise.

OWASP Top 10 Threats in 2010	Key Considerations
A1: Injection flaws	Separate un-trusted data from user-supplied application command or query. **Who can send data** to systems?
A2: Cross-site scripting (XSS)	Separate un-trusted data from active browsers. **Who can send data** to systems?
A3: Broken authentication & session management	Need to **control access** with the ability to invalidate session state at logout. No reuse of tokens or SSL state should be allowed.
A4: Insecure direct object reference	Do any users have **partial access** to change system data?
A5: Cross-site request forgery (CSRF)	Need to **control access** with ability to deny,"step-up," or re-authenticate the user.
A6: Security misconfiguration	Have you performed **security hardening** across the entire application stack?
A7: Insecure cryptographic storage	**Encrypt** sensitive data. Use security tokens to protect cryptographic resources.
A8: Failure to restrict URL access	Need to **control access** to URLs on the portal. Can anyone with network access send an application request?
A9: Insufficient transport layer protection	Can anyone monitor the network traffic of your users? **Use SSL** to protect all authenticated traffic.
A10: Unvalidated redirects & forwards	Can anyone **trick your users** into submitting a request to your website?

Table 11: OWASP Top 10 Threat list for 2010

Web application platforms and vulnerabilities

Counting vulnerabilities for Web application platforms is a bit more complex than counting them for regular Web applications. When analyzing Web application platforms, we find it useful to distinguish between the base platform and any plug-ins that the Web application platform uses. Plug-ins may or may not be produced by the Web application vendors themselves. While plug-ins extend the functionality of many of these Web application platforms, they may not be as rigorously coded or as quickly updated as the platforms they support. More importantly, the plug-ins are where the vast majority of vulnerabilities occur.

Figure 44 shows the percentage of all vulnerability disclosures in the first half of 2010 for the main Web application platforms and their plug-ins. We include only those Web application platforms and their associated plug-ins with 10 or more disclosed vulnerabilities.

Taken together, the major Web application platforms and their plug-ins account for almost 14 percent of all vulnerability disclosures reported in the first half of 2010.

Observe that the clear majority of disclosed vulnerabilities related to Web application platforms are for plug-in (88 percent) versus the Web application platforms themselves (12 percent).

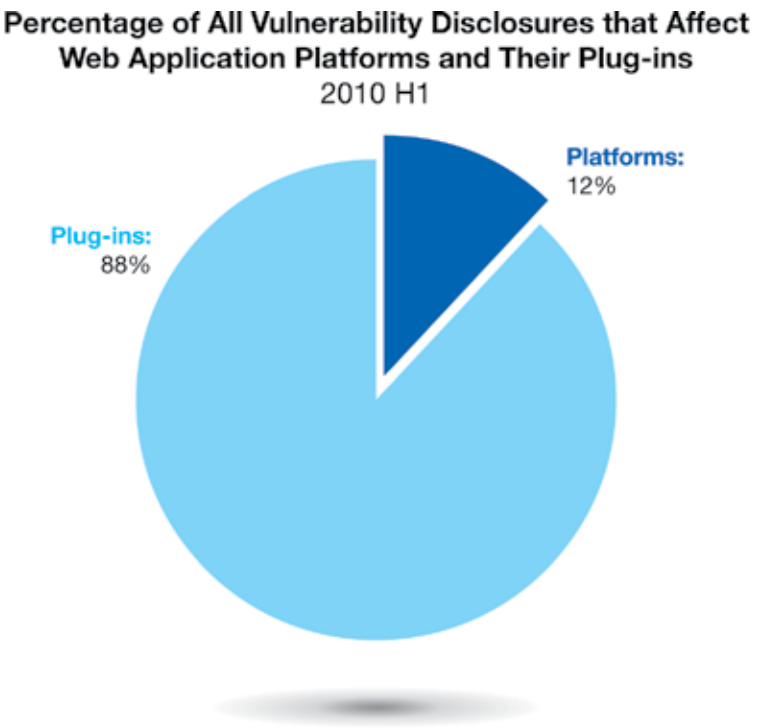


Figure 44: Percentage of all vulnerability disclosures that affect Web application platforms and their plug-ins, 2010 H1

The following graph shows the relative counts of disclosed vulnerabilities for plug-ins and Web application platforms.

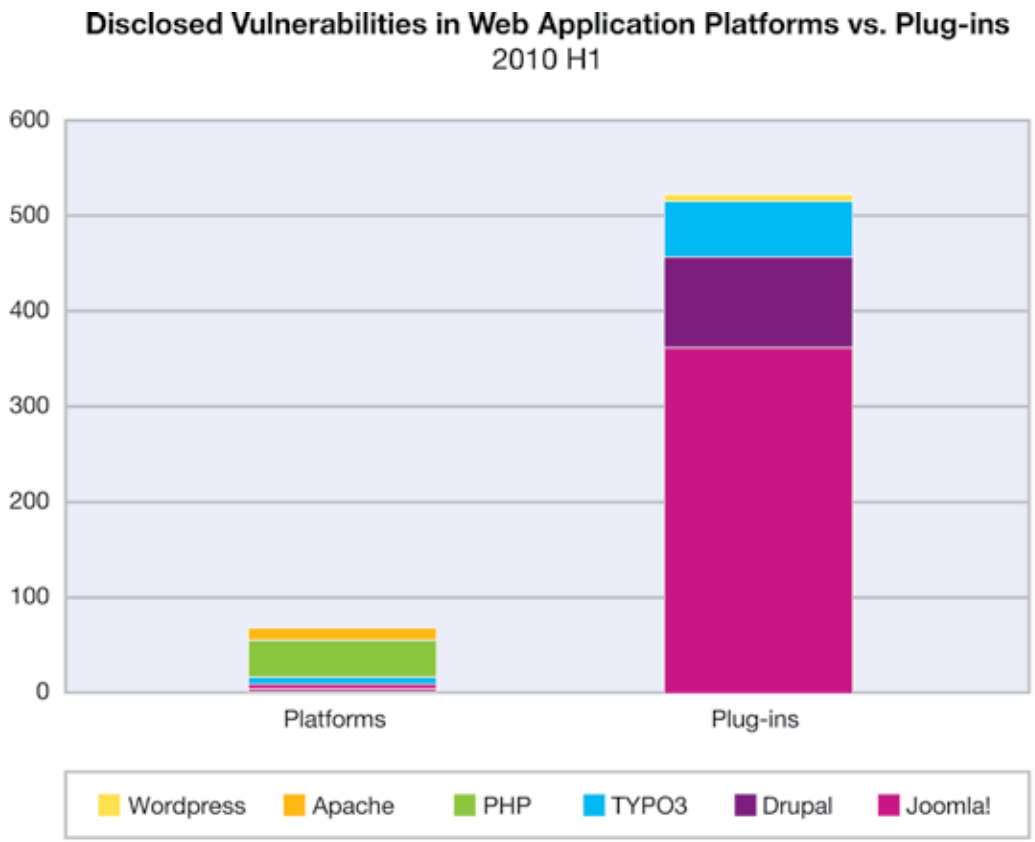


Figure 45: Disclosed vulnerabilities in Web application platforms versus plug-ins, 2010 H1

What can we learn from this?

Although Web application vendors have very few vulnerabilities that are attributed to the code that they produce, if your organization relies heavily on the many plug-ins provided to support these applications, spend some time to investigate and remediate any disclosed vulnerabilities. Better yet, fully assess the finished product with a Web application scanner before deployment to ensure that no undisclosed vulnerabilities exist or were introduced during the development process. Ensuring that these applications are safe before they are deployed will help prevent your website from becoming a springboard for attackers.

Browser and other client-side vulnerabilities and exploits

Prevalent client-side software—percent of critical and high vulnerability disclosures

Looking back to our 2009 end of year report, client-side vulnerabilities declined by five percent compared to 2008. Still, these vulnerabilities, which affect personal computers, continued to represent the second-largest category of vulnerability disclosures after Web application vulnerabilities and represented about a fifth of all vulnerability disclosures.

Figure 46 represents the current breakdown of critical and high categories when looking across the various client-side applications.

Even though in the mid-year we see a decline in the chart ending in 2009, we remind you that this is only six-month data. In the first half of 2010, we see that document readers and editors, as well as multimedia applications, have almost surpassed 2009 year-end totals. Browser applications have clearly hit the half-way point for this year and are expected to continue. In keeping with the high number of vulnerability disclosures for 2010, we expect these areas to hit record numbers by the end of the year.

As we have discussed in earlier sections, these record-setting numbers of disclosures are being actively utilized by attackers and represent significant security issues that we will continue to watch.

The major types of vulnerabilities affecting clients continue to fall into one of four main categories shown in Table 12.

Category	Description
Browser	Client Web browser software and plug-ins.
Document Reader and Editor	Software that allows users to create or view documents, spreadsheets, presentations, and other types of files that are not images, music, or movies.
Multimedia	Software that allows users to view or create music and movies.
Operating System	The base operating system, excluding applications that are in the other three categories.

Table 12: Key vulnerability categories related to client-side vulnerability disclosures

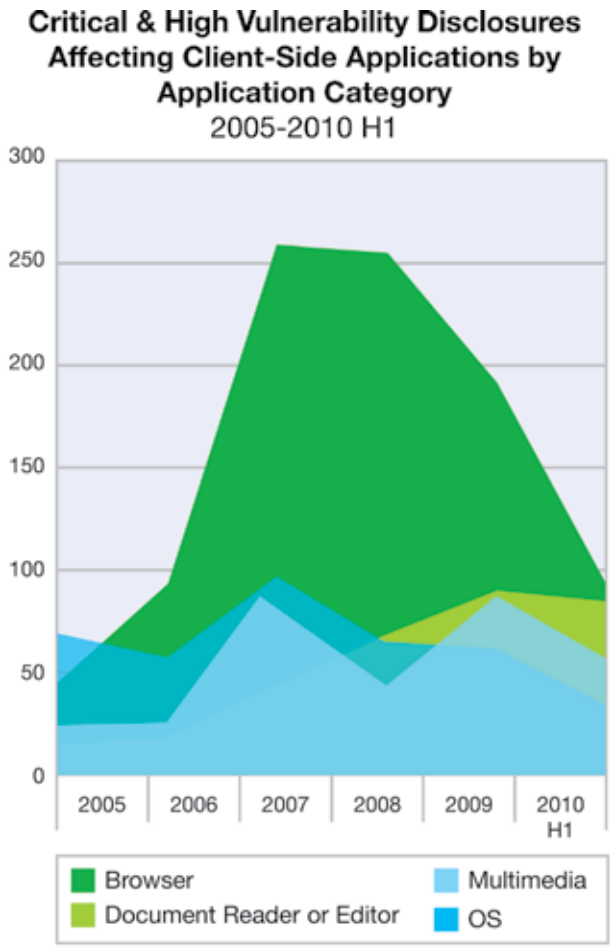


Figure 46: Critical & high vulnerability disclosures affecting client-side applications by application category, 2005-2010 H1

Browser vulnerabilities—Internet Explorer surges ahead in 2010

The largest category of client-side vulnerabilities remains the browser category. This category includes not only the browsers themselves but the many plug-ins that can be installed on browsers. We continue to see decline in affected ActiveX controls.

As we would expect from the increase in disclosures this year, both Mozilla Firefox and Microsoft Internet Explorer are experiencing increased numbers in the first half of 2010. At this point both browsers are relatively parallel with disclosures although it appears today that Internet Explorer has already reached easily two-thirds of the total 2009 numbers reported for the browser.

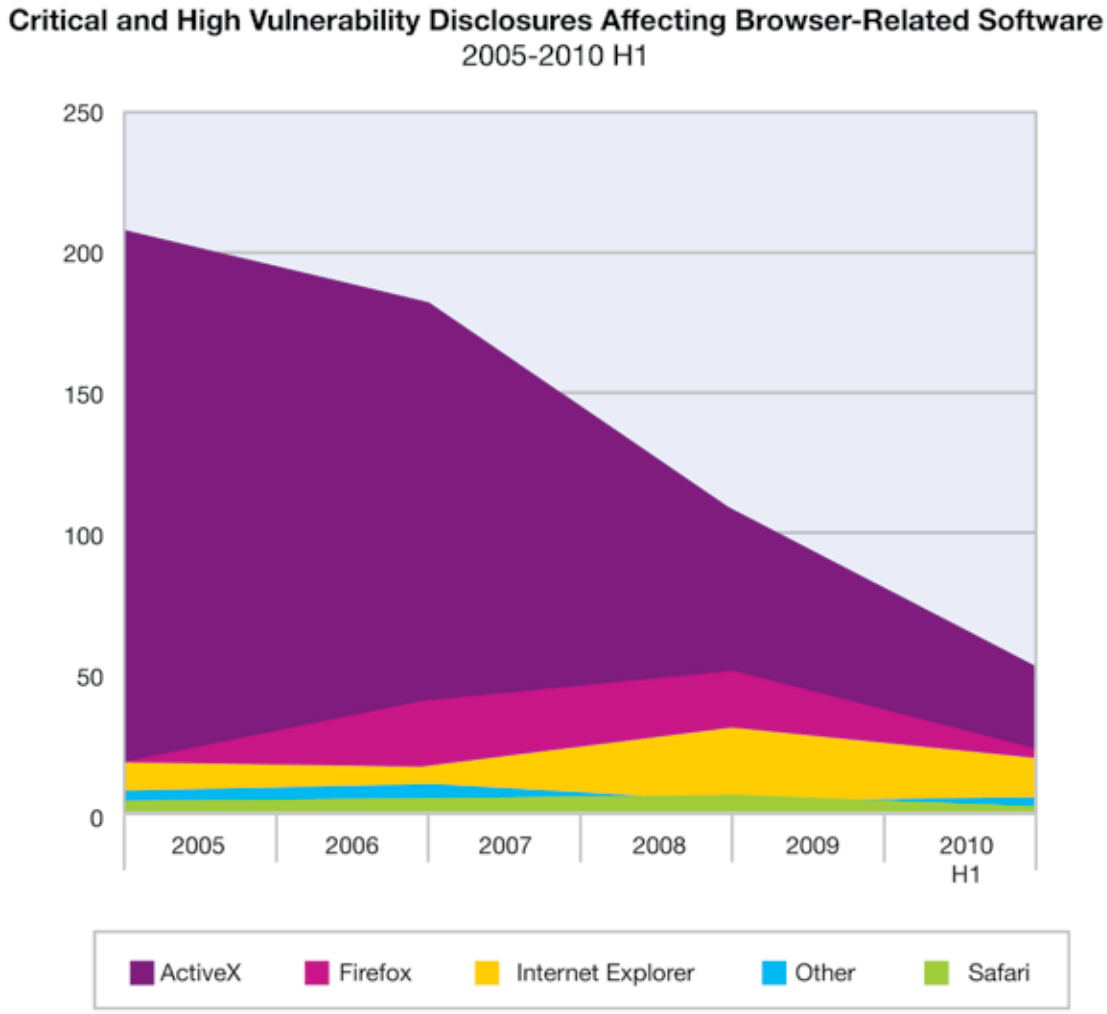


Figure 47: Critical and high vulnerability disclosures affecting browser-related software, 2005-2010 H1

Document format vulnerabilities

When it comes to document vulnerabilities, two predominant types of document vulnerabilities are evident: Office documents and Portable Document Format (PDF) documents.

PDF'S at this mid-year point now represent roughly 49% of the 2009 end of year totals. We see below in Figure 48 the ongoing trend where office document vulnerability is declining and PDF continues to rise.

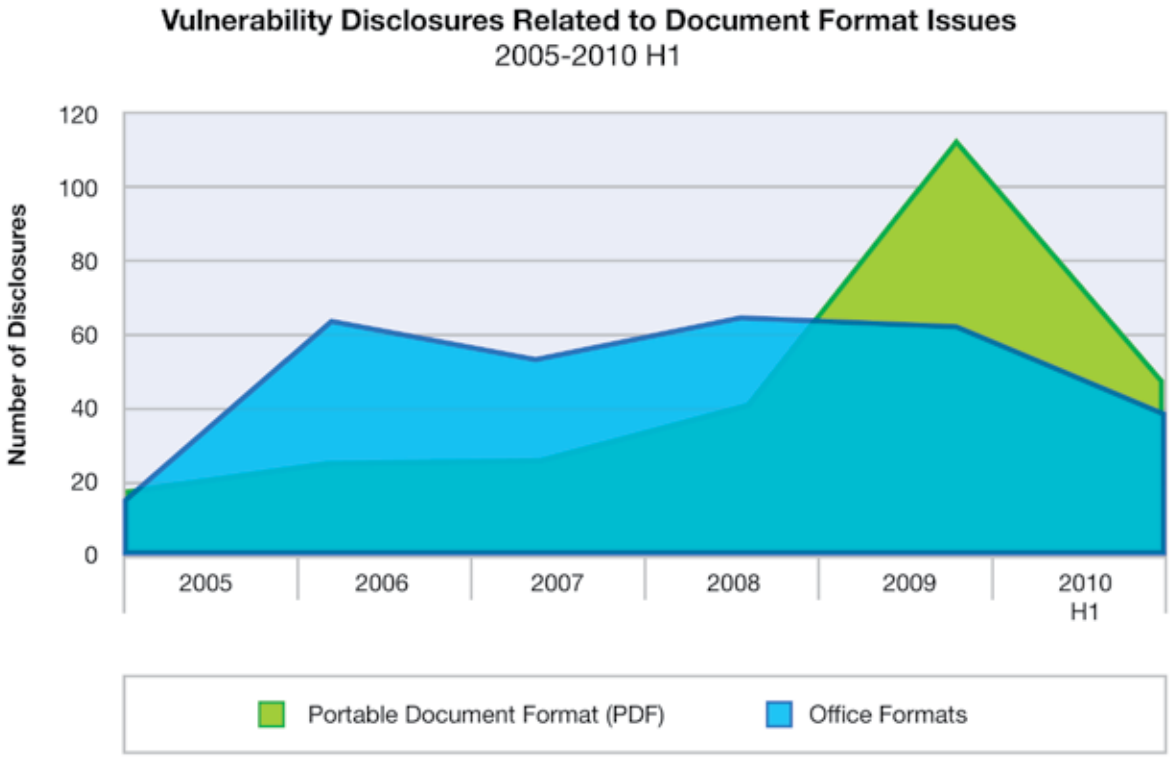


Figure 48: Vulnerability disclosures related to document format issues, 2005-2010 H1

As we look across the threat landscape, we have reported that PDF has become the weapon of choice for many attackers. Our sensors detected a round of attacks in February of this year involving spam emails with obfuscated PDF attachments that exploited vulnerabilities in Acrobat to install malware. Our researchers **blogged** about these attacks, which combined different methods of evading modern AV and Spam filters. In April, our sensors detected another **significant surge** in malicious PDF spam. This round used the Javascript Launch command to install the Zeus botnet on victim's computers.

We remind security specialists to stay vigilant of these methods and to educate your users to the threats these documents pose.

Client exploitation trends

X-Force monitors client exploits through several projects and services.

- IBM Managed Security Services (MSS), responsible for monitoring exploits related not only to endpoints, but also servers (including Web servers) and general network infrastructure. This data tracks exploits delivered over the Web in addition to other vectors like email and instant messaging.
- Our "Whiro" crawlers, which combine alert data from MSS, our "C-Force", and independent analysis to monitor exploitation from Web-based sources. Whiro uses specialized technology to identify exploits used even in the most obfuscated cases including where toolkits attempt multiple exploits.
- Our Content team independently scours and categorizes the Web through crawling, independent discoveries, and through the feeds provided by MSS and Whiro.

Web browser exploitation trends

X-Force continues to track growth in Web browser exploitation through its Whiro crawlers and analysis of IBM Managed Security Services operational alert data. When we decided to produce a similar statistic for Web exploit toolkit prevalence, we learned that it was both possible and tricky to accomplish due to code similarities and coded theft. Over time, we have also been tracking trends in obfuscation techniques used by individual exploits as well as in the Web browser exploit toolkits. We continue to invest in new technologies to improve our results in this domain.

Just as lone Web browser exploit sites in the wild are dying and exploit toolkits and groups are taking the forefront of Web browser exploitation, we are seeing some disturbing new possibilities in anti-analysis. Exploit kits in the wild are denying serving content more than once to a particular Internet Protocol (IP) address in greater numbers. This feature has two practical benefits to the attacker: 1) that the infection only happens once to avoid potential destabilization of the victim and 2) to hinder analysis. This filtering approach is not exactly new and neither is referrer checking but referrer checking is making a comeback. By blocking requests without a valid referrer, such as the URL of a compromised Webpage or malicious advertisement, just sharing malicious URLs is not e enough to obtain the malicious sample(s).

Most popular exploits (2010 H1)

1. CVE-2007-5659, PDF Collab.CollectEmailInfo
2. CVE-2009-0927, PDF getIcon
3. CVE-2008-2992, PDF Util.Printf
4. CVE-2007-0071, SWF Scene Count
5. CVE-2008-5353, Java Object Deserialization

The continued prevalence of Gumblar—the exploit toolkit/group—is still helping to secure top positions for Adobe products, but PDF and Flash exploits are extremely popular in many other exploit toolkits as well. An interesting change from the second half of 2009 is that ActiveX has dropped off the top-five list, at least for now. In our 2009 full-year report, we projected that Adobe products would continue to be major players on this list, but without committing to whether it was going to be dominated by PDF or Flash. Judging by what we have observed thus far in 2010, it is safe to assume that 2010 will be dominated by PDF exploitation.

Additionally, an older Java vulnerability has found its way onto our top-five list at the bottom position. Considering that Java, PDF, and Flash are consistent across different browser environments, it is clear that attackers are interested in catching users of non-IE browsers without needing to invest time or money in browser-specific attacks. Attackers further leverage vendor patch cycles that have decreased over time for key browser vendors but not necessarily for browser and cross-browser plug-in vendors. In 2010, X-Force predicts that browser plug-in vendors will begin addressing zero-day attacks much more quickly than in the past. However, even when patches reach the market faster, it doesn't help if computer users do not use auto-update features, notifications, or rigorously manage their patch state manually.

Most popular exploit toolkits (2010 H1)

1. Gumblar
2. Fragus
3. Eleonore
4. Phoenix
5. JustExploit

Tabulating exploit kit prevalence involves numerous challenges. The long-standing issue of code similarity and kit branches with unique obfuscation makes this task difficult. Our approach to determining prevalence is based on heuristics applied to malicious content which is de-obfuscated. While X-Force is generally confident in these results, we admit that our exploit crawler has some difficulty with the latest Neosploit kits. We estimate that Neosploit should fall between Eleonore and Phoenix, causing JustExploit to drop off of the list. Looking back at our mid-year and full-year 2009 results for most popular exploit toolkits, we see that Gumblar continues to dominate the results.

A newer kit, Fragus, has rocketed to second place, while Eleonore has gone from fifth place in the second half of 2009 to third place during the first half of 2010. Phoenix continues to have a position on our list, although it is carried over from the full-year rather than second-half results. Avid readers of our trend reports may notice that there is a decent amount of churn in terms of the toolkits that attackers favor over time. Therefore, trying to predict full-year 2010 results is tricky. X-Force thinks Gumblar will still exist as the number one toolkit, and as mentioned previously, also expects JustExploit to drop off the list. However, the other three may continue going strong or one or two may fizzle depending on what other toolkits appear.

Web content trends

This section summarizes the amount and distribution of "bad" Web content that is typically unwanted by businesses based on social principles and corporate policy. Unwanted or "bad" Internet content is associated with three types of websites: the IBM Web filter categories that correspond with these types of sites.

The Web filter categories are defined in detail at: **http://www-935.ibm.com/services/us/index.wss/detail/iss/a1029077?cntxt=a1027244**

This section provides analysis for:

- The percent and distribution of Web content that is considered bad or unwanted
- The increase in the amount of anonymous proxies
- Web pages with links to malware URLs

Website Type	Description & Web Filter Category
Adult	Pornography Erotic / Sex
Social Deviance	Political Extreme / Hate / Discrimination Sects
Criminal	Anonymous Proxies Computer Crime / Hacking Illegal Activities Illegal Drugs Malware Violence / Extreme Warez / Software Piracy

Table 13: Web filter categories associated with unwanted Web content

Analysis methodology

X-Force captures information about the distribution of content on the Internet by counting the hosts categorized in the IBM Security Solutions Web filter database. Counting hosts is an accepted method for determining content distribution and provides the most realistic assessment. When using other methodologies—like counting Web pages and sub pages—results may differ.

The IBM Content data center constantly reviews and analyzes new Web content data. The IBM Content data center analyzes 150 million new Web pages and images each month and has analyzed 13 billion Web pages and images since 1999.

The IBM Web Filter Database has 68 filter categories and 65 million entries with 150,000 new or updated entries added each day.

Percentage of unwanted Internet content

Approximately 7.2 percent of the Internet currently contains unwanted content such as pornographic or criminal websites.

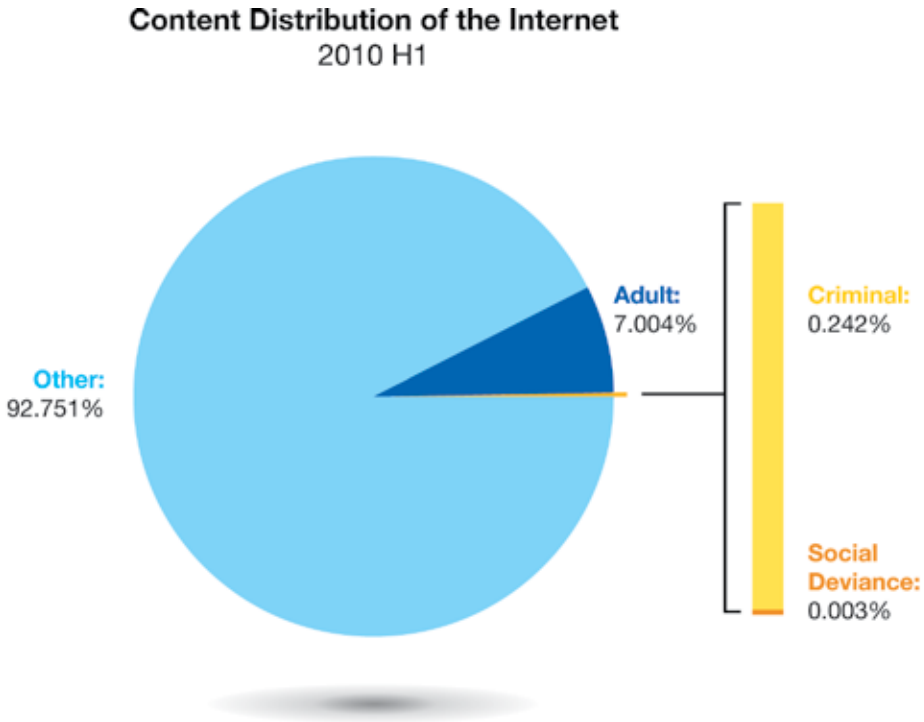


Figure 49: Content distribution of the Internet, 2010 H1

Increase of anonymous proxies

As the Internet becomes a more integrated part of our lives, not only at home but also at work and school, organizations responsible for maintaining acceptable environments increasingly find the need to control where people can browse in these public settings.

One such control is a content filtering system that prevents access to unacceptable or inappropriate websites. Some individuals attempt to use anonymous proxies (also known as Web proxies) to circumvent Web filtering technologies.

Web proxies allow users to enter an URL on a Web form instead of directly visiting the target website. Using the proxy hides the target URL from a Web filter. If the Web filter is not set up to monitor or block anonymous proxies, then this activity (which normally would have been stopped) bypasses the filter and allows the user to reach the disallowed website.

The growth in volume of anonymous proxy websites shown in Figure 50 reflects this trend.

In the past three years, anonymous proxies have steadily increased, more than quadrupling in number. Anonymous proxies are a critical type of website to track, because of the ease at which proxies allow people to hide potentially malicious intent.

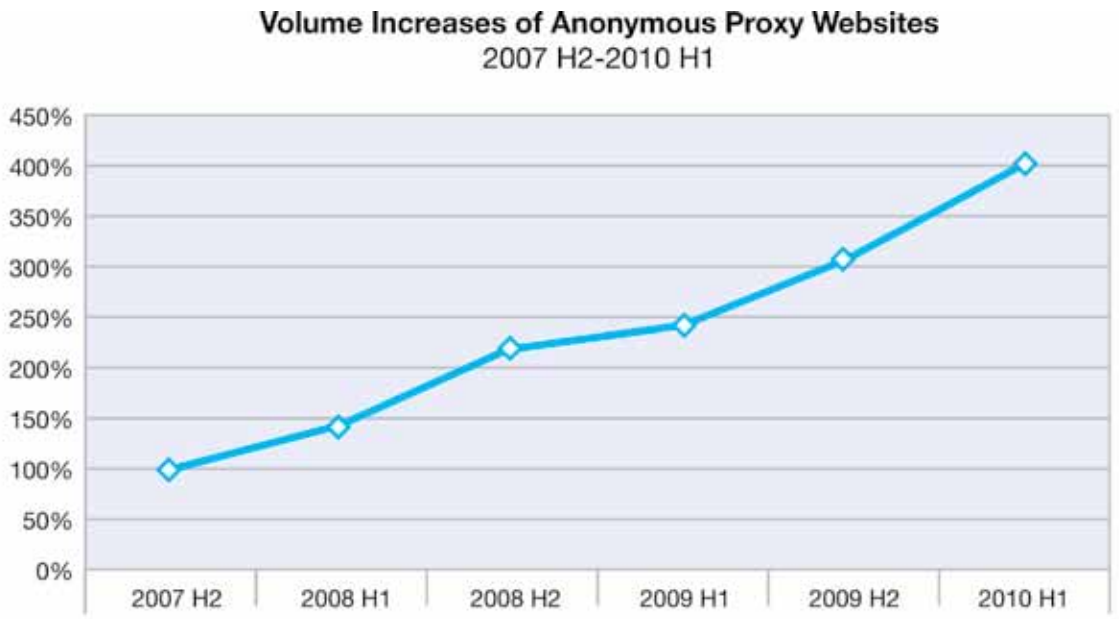


Figure 50: Volume increases of anonymous proxy websites, 2007 H2-2010 H1

Top level domains of anonymous proxies

Figure 51 illustrates the Top Level Domains (TLDs) of newly registered anonymous proxies.

In 2006, more than 60 percent of all newly-registered anonymous proxies were **.com** domains, but since the middle of 2007, **.info** has been at the top until the beginning of 2010 (while **.com** was runner-up for most of the time).

But why is **.info** no longer in the prime position? It seemed to be a proven TLD for anonymous proxies for years. A reason could be that .info, similar to **.com**, is running out of names. So the question arises why are anonymous proxies now provided on **.cc** and **.tk** top level domains?

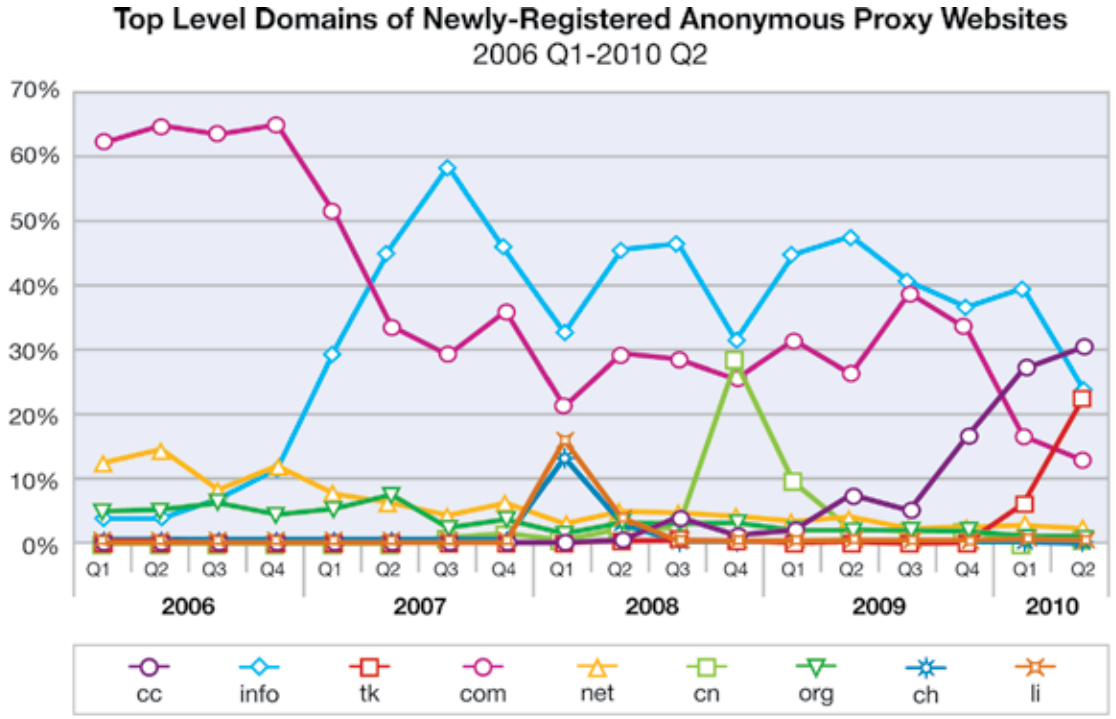


Figure 51: Top level domains of newly-registered anonymous proxy websites, 2006 Q1-2010 Q2

These are the domains of Cocos (Keeling) Islands (**.cc**), an Australian territory and Tokelau (**.tk**), a territory of New Zealand. The domain **.cc** is administered by VeriSign. Nearly all **.cc** anonymous proxy websites are registered on the domain **.co.cc**. It is free of charge to register a domain anything **.co.cc** (see http://www.co.cc/?lang=en). The same is true for **.tk**. (see http://www.dot.tk/). Thus, it is very cheap and attractive to install new anonymous proxies on **.co.cc**, or **.tk**.

Additional trends:

- At the beginning of 2008, the top level domains of neighboring countries Switzerland (**.ch**) and Liechtenstein (**.li**) together represented about 30 percent of the newly registered anonymous proxies.
- In the fourth quarter of 2008, the top level domain of China reached nearly 30 percent of the newly registered anonymous proxies.
- At the end of 2009 **.cc** (Cocos (Keeling) Islands) started to increase significantly and even reached the number one position in the second quarter of 2010.
- In the second quarter of 2010, another new star in proxy heaven, **.tk** (Tokelau), reached about 23 percent of new anonymous proxies.
- During that same time period, **.info** decreased dramatically and fell below 30 percent for the first time since beginning of 2007.
- In the first quarter of 2010, even **.com** fell below 20 percent for the first time.

Country hosts of anonymous proxy websites

For anonymous proxy hosting countries, the United States has held the top position for years. More than 70 percent of all newly registered anonymous proxies have been hosted in the US over the last four and a half years. This percentage climbed to more than 80 percent from the middle of 2008 until the end of 2009. During the first half of 2010, about 75 percent of all newly-registered anonymous proxies were hosted in the US.

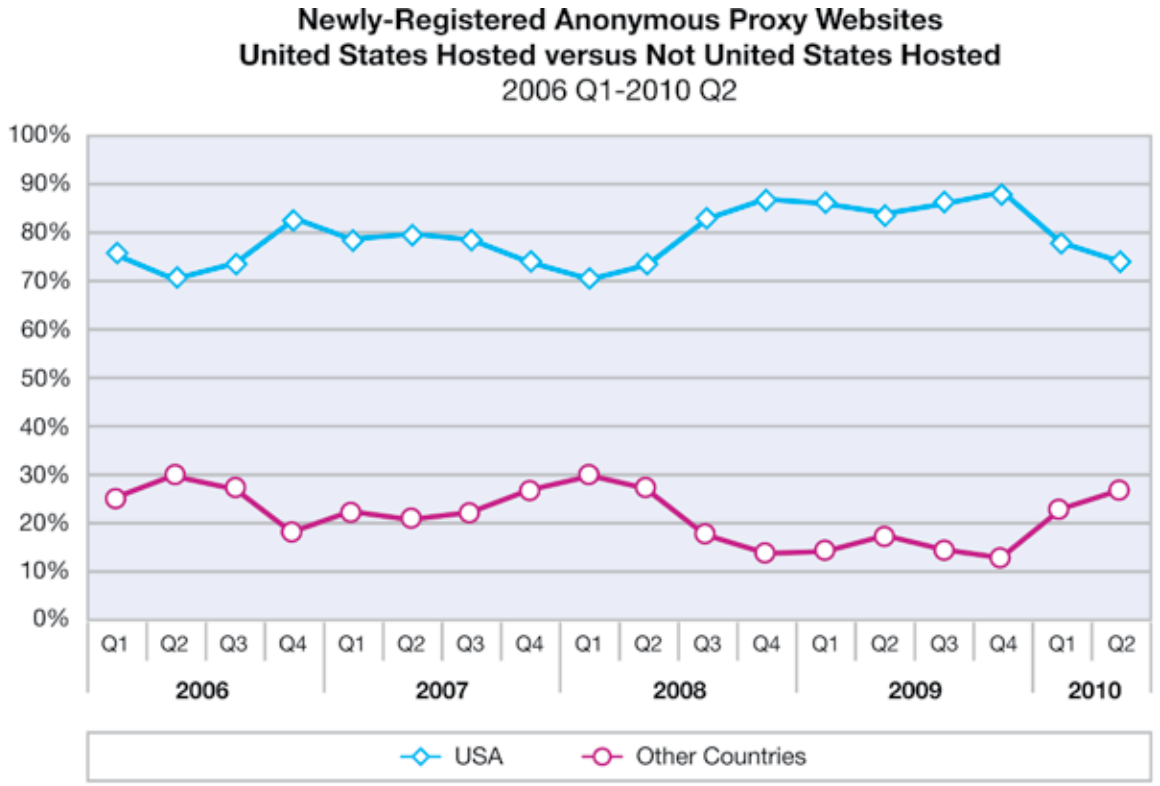


Figure 52: Newly-registered anonymous proxy websites United States hosted versus not United States hosted, 2006 Q1-2010 Q2

It is worth looking at the remaining 25 percent of all newly registered anonymous proxies in the first half of 2010. This remainder is dominated by Canada in the first quarter (7.9 percent) and UK in the second quarter (7.8 percent). All other countries host less than 4 percent thus far in 2010.

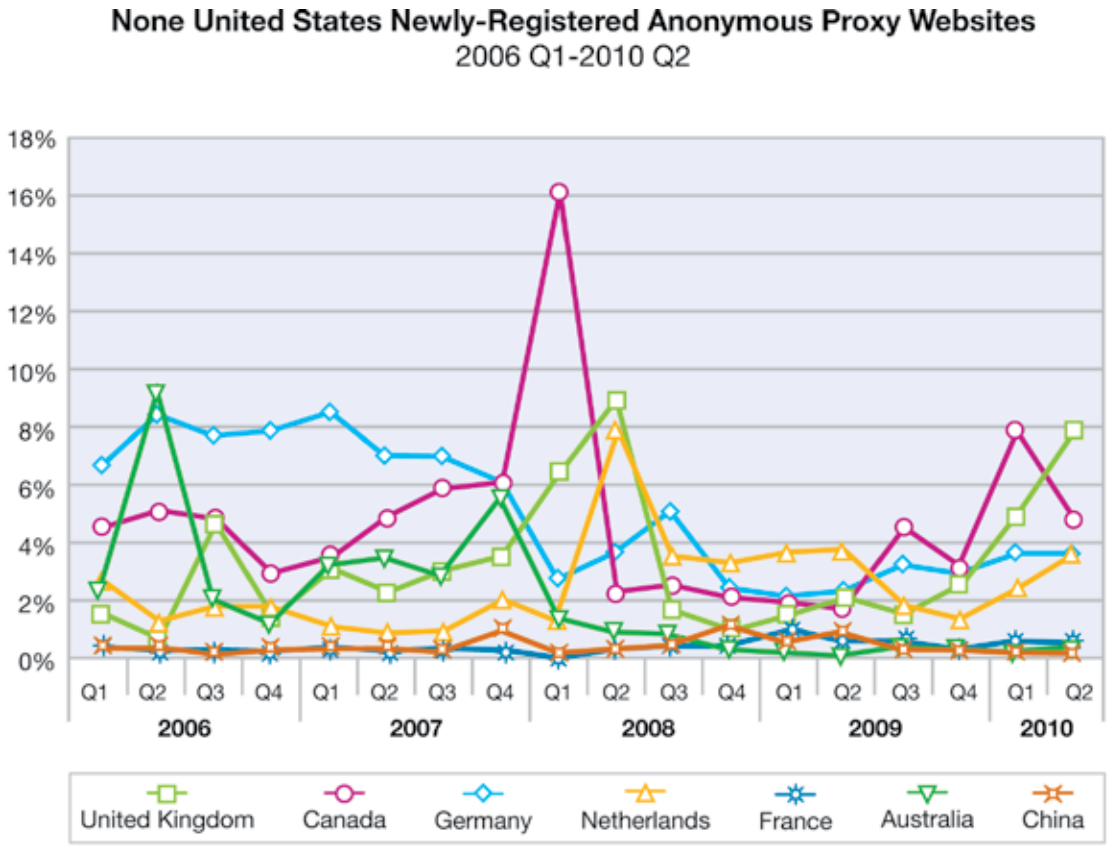


Figure 53: None United States newly-registered anonymous proxy websites, 2006 Q1-2010 Q2

Good websites with bad links

As described in Web Application Threats and Vulnerabilities on page 71 and in Common domains in URL spam on page 94, attackers are focusing more and more on using the good name of trusted websites to lower the guard of end users and attempt to obfuscate their attempts with protection technologies. The use of malicious Web content is no different. The following analysis provides a glimpse into the types of websites that most frequently contain links to known, malicious links.

Some of the top categories might not be surprising. For example, one might expect pornography to top the list. It does, and it has gotten worse over the past 12 months. However, the second-tier candidates fall into the more "trusted" category.

Blogs, bulletin boards, personal websites, search engines, education, online magazines, and news sites fall into this second-tier category. Most of these websites allow users to upload content or design their own website, such as personal content on university's site or comments about a "purchase" on a shopping website. In other words, it is unlikely that these types of websites are intentionally hosting malicious links. The distribution is probably more representative of the types of websites that attackers like to frequent in hopes of finding a loop-hole (like a vulnerability or an area that allows user-supplied content) in which they can incorporate these malicious links in hopes of compromising an unsuspecting victim.

Figure 54 lists the most common types of websites that host at least one link that points back to a known malicious website.

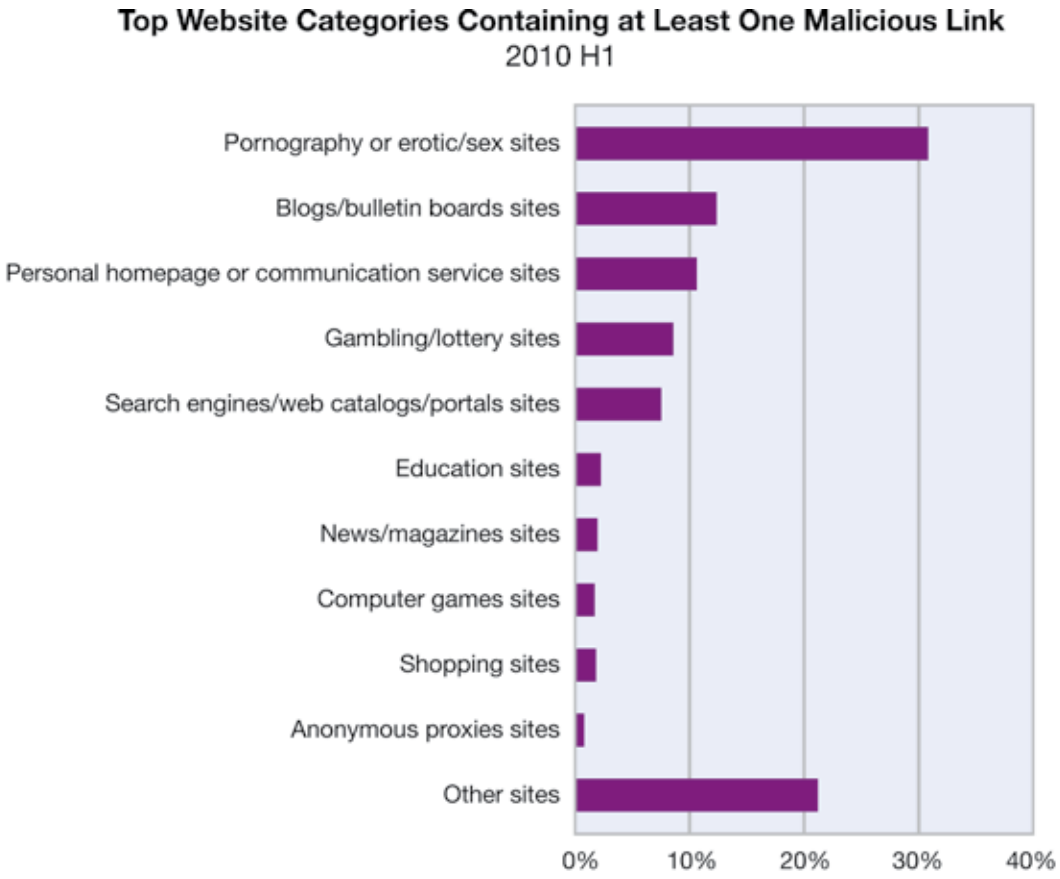


Figure 54: Top website categories containing at least one malicious link, 2010 H1

Referring to Figure 55, some interesting trends appear when we compare current data to the data seen six and even 12 months ago. Professional "bad" websites like pornography or gambling websites have increased their links to malware, making it appear more likely that "professionals" are improving their efforts to systematically distribute their malware.

Blogs and bulletin boards, too, have seen increases in malware links. This is likely due to increased infiltration by attackers without adequate controls set in place by blog and bulletin board owners. This trend has slowed over the last six months but we have noticed increases for computer games and anonymous proxy sites.

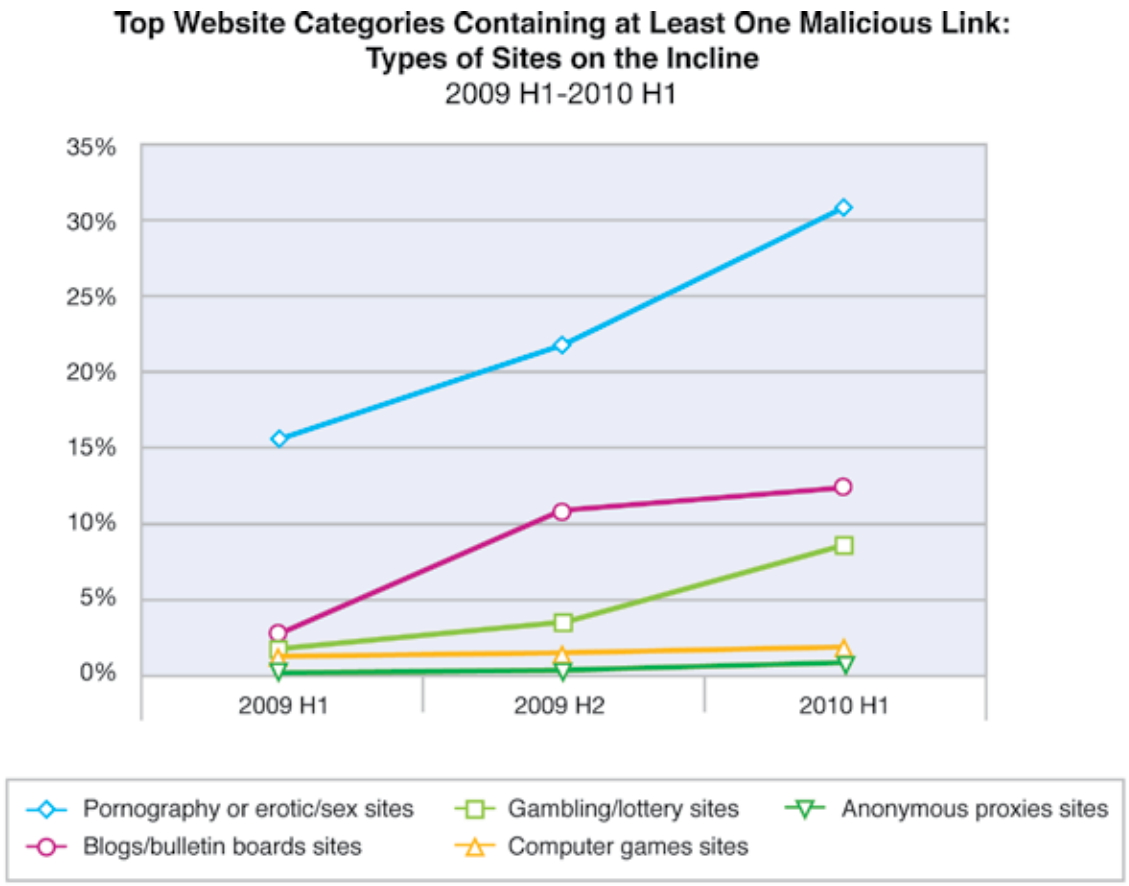


Figure 55: Top website categories containing at least one malicious link: types of sites on the incline, 2009 H1-2010 H1

Personal home pages are no longer the most prevalent category that host at least one malicious link. Personal home pages have improved in comparison to the first half of 2009. One reason may be that personal homepages are more out of style in favor of Web 2.0 applications like profiles in social or business networks. Search engines, portals, shopping sites, education, and news sites have also improved. These "traditional" legitimate interactive sites have been used to exchange information and opinions for years. Thus, it is likely that providers of those services have increased their efforts in IT security.

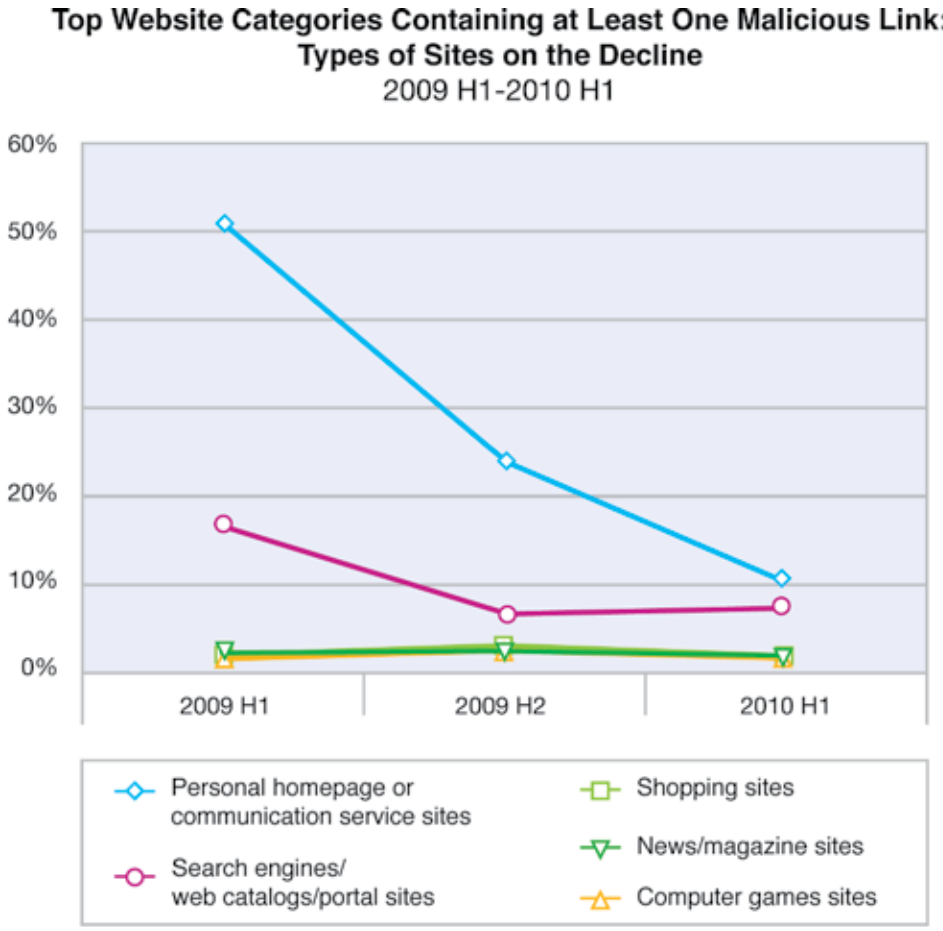


Figure 56: Top website categories containing at least one malicious link: types of sites on the decline, 2009 H1-2010 H1

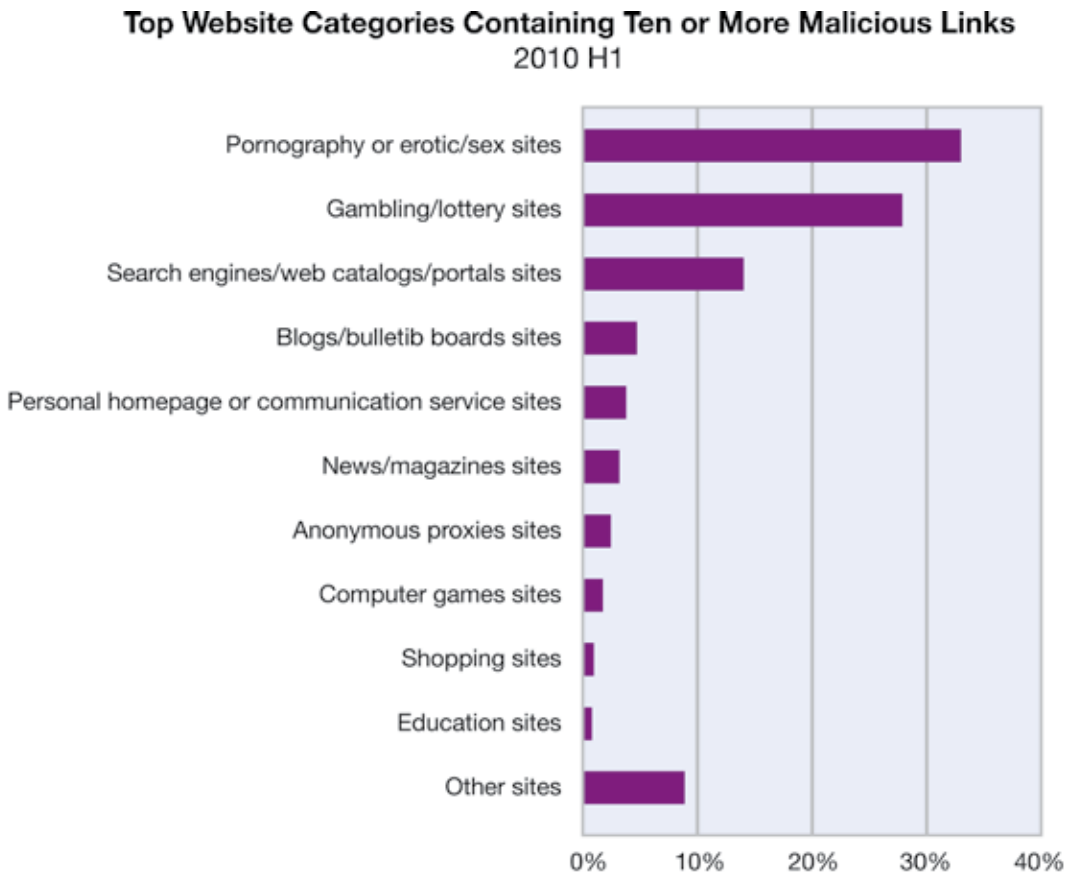


Figure 57: Top website categories containing ten or more malicious links, 2010 H1

Another way to look at this problem is to examine websites that appear to be hosting an extraordinary number of links to malicious websites. When you do an analysis of those sites that host 10 or more malicious links, another story emerges—one that implies that the owners of some of these websites may be enjoying the financial advantage that these compromises provide. Out of the categories of websites that host 10 or more of these links, pornography accounts for nearly 33 percent and gambling accounts for about 28 percent. One might suspect that these kinds of websites knowingly use these links for profit. Some appear to have links placed systematically throughout the site.

Compared to the data six months ago, the values in most categories have changed by four percent or less. But pornography sites gained by six percent and gambling sites gained by 11.4 percent. Hence, malware distributors focus more and more on these popular but dark sites of the Internet. Against the background of 0.6 percent of the adult population having problem gambling issues (see **http://en.wikipedia.org/wiki/Gambling_addiction#Prevalence**), gambling sites are a popular target for malware distributors.

Spam

The IBM spam and URL filter database provides a world-encompassing view of spam and phishing attacks. With millions of email addresses being actively monitored, the content team has identified numerous advances in the spam and phishing technologies that attackers use.

Currently, our spam and URL filter database contains more than 40 million relevant spam signatures. Each piece of spam is broken into several logical parts (sentences, paragraphs, etc.). A unique 128-bit signature is computed for each part and for millions of spam URLs. Each day there are approximately one million new, updated, or deleted signatures in the spam filter database.

The topics included in this section are:

- Spam volume
- New trends in types of spam
- Most popular domains used in spam
- Most popular Top Level Domains (TLDs) used in spam and why the top domains are so popular
- Reputation of Spam URLs
- Spam's country[1] of origin trends, including spam Web pages (URLs)
- Changes in the average byte size of spam
- Most popular subject lines of spam

Spam volume

In the beginning of 2009, spam volume stagnated for a couple of months. In May 2009, spam volume started to increase and over time surpassed the spam level seen just before the **McColo shutdown**. In the fourth quarter of 2009, spammers started a year-end rally. In November, they sent out twice as much spam than before the McColo shutdown. In 2010 spammers maintained a steady level until April when the amount of spam began to increase again, finally reaching an all-time high in June.

Figure 58: Changes in spam volume, April 2008-June 2010

[1] The statistics in this report for spam, phishing, and URLs use the IP-to-Country Database provided by **WebHosting.Info**, available from **http://ip-to-country.Webhosting.info**. The geographical distribution was determined by requesting the IP addresses of the hosts (in the case of the content distribution) or of the sending mail server (in the case of spam and phishing) to the IP-to-Country Database.

Types of spam

Over the years spammers have focused on using the most unsuspicious type of email, HTML-based spam without attachments. The chart below shows a significant increase in this type of spam until the beginning of 2009 while plain-text spam (without other email parts or attachments) decreased in the same time period.

Since the second quarter of 2009, HTML spam is ranging between 81 and 84 percent. In the second and third quarter of 2009, we witnessed a short rebirth of image-based spam. Plain-text spam increased from late 2009 into early 2010. At the same time, image-based spam decreased and did not play a major role in the first half of 2010.

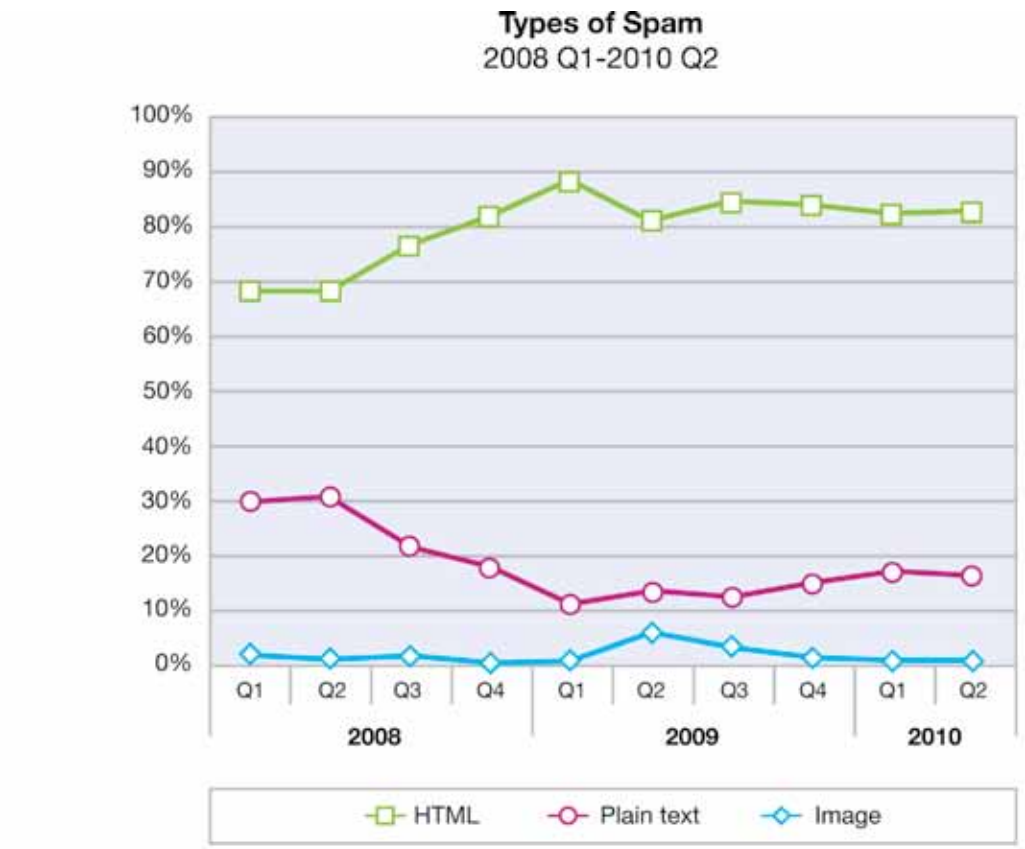


Figure 59: Types of spam, 2008 Q1-2010 Q2

Common domains in URL spam

The vast majority of spam, more than 90 percent, is classified as URL spam-spam messages that include URLs that a person clicks to view the spam contents.

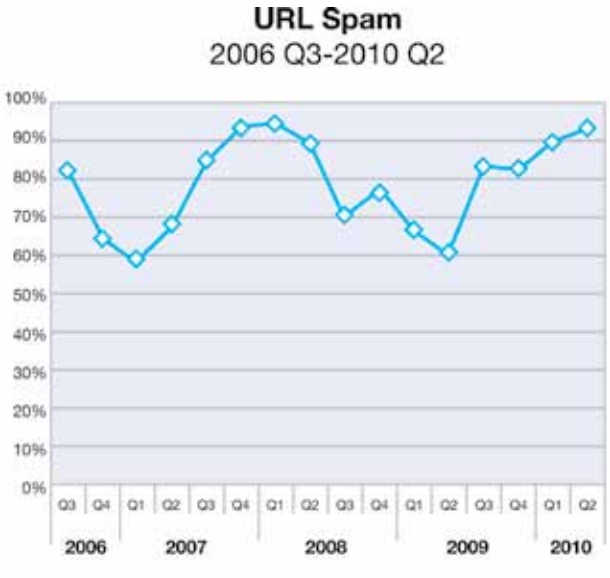


Figure 60: URL spam, 2006 Q3-2010 Q2

It is worthwhile to take a closer look at the most frequently used domain names in URL spam. Table 14 shows the top 10 domains per month for the first half of 2010. We have highlighted domains that are well known or have been registered for a long time and are not registered for hosting spam content.

Rank	January 2010	February 2010	March 2010	April 2010	May 2010	June 2010
1.	flickr.com	radikal.ru	livefilestore.com	livefilestore.com	imageshack.us	imageshack.us
2.	imageshack.us	imageshack.us	imageboo.com	imageshack.us	imageshost.ru	imageshost.ru
3.	radikal.ru	livefilestore.com	radikal.ru	imageshost.ru	myimg.de	pikucha.ru
4.	livefilestore.com	flickr.com	imageshack.us	imgur.com	xs.to	imgur.com
5.	Webmd.com	live.com	googlegroups.com	myimg.de	imgur.com	myasvir.com
6.	picsochka.ru	imageboo.com	live.com	xs.to	tinypic.com	mojoimage.com
7.	live.com	capalola.biz	akamaitech.net	icontact.com	livefilestore.com	myimg.de
8.	superbshore.com	feetorder.ru	gonestory.com	tinypic.com	icontact.com	twimg.com
9.	tumblr.com	laughexcite.ru	bestanswer.ru	live.com	googlegroups.com	icontact.com
10.	fairgreat.com	hismouth.ru	wrotelike.ru	binkyou.com	images-amazon.com	twitter.com

Table 14: Most common domains in URL spam, 2010 H1

The majority of these domain names are well known and trusted, continuing the trend of the last few years. Figure 61 shows the percentage of spam domains versus trusted domains within the top ten domains used for spam from the first half of 2008 through the first half of 2010.

Some of the well-known websites include:

- **akamaitech.net** (website of Akamai Technologies)
- **googlegroups.com** (free service from Google where groups of people can discuss common interests)
- **icontact.com** (email marketing offering company)
- **images-amazon.com** (domain owned by Amazon.com, Inc.)
- **live.com** (a Windows Live service that allows users to create a personalized homepage)
- **livefilestore.com** (Microsoft's Web Storage service)
- **tumblr.com** (blogging platform)
- **twimg.com** (domain owned by Twitter)
- **twitter.com** (Twitter website)
- **Webmd.com** (official website of WebMD Health Corporation, an American provider of health information services)

Major targeted image-hosting websites were:

- **flickr.com** (official website of Flickr)
- **imageshack.us** (official website of ImageShack)

And there are also some smaller- and medium-sized image-hosting websites:

- **imageboo.com**
- **imageshost.ru**
- **imgur.com**
- **mojoimage.com**
- **myimg.de**
- **mytasvir.com**
- **pikucha.ru**
- **radikal.ru**
- **tinypic.ru**
- **xs.to**

Not only do the above websites provide recognizable (and trustworthy) Web links for the end user, but spam messages can successfully evade some anti-spam technology by using these legitimate links in their spam emails.

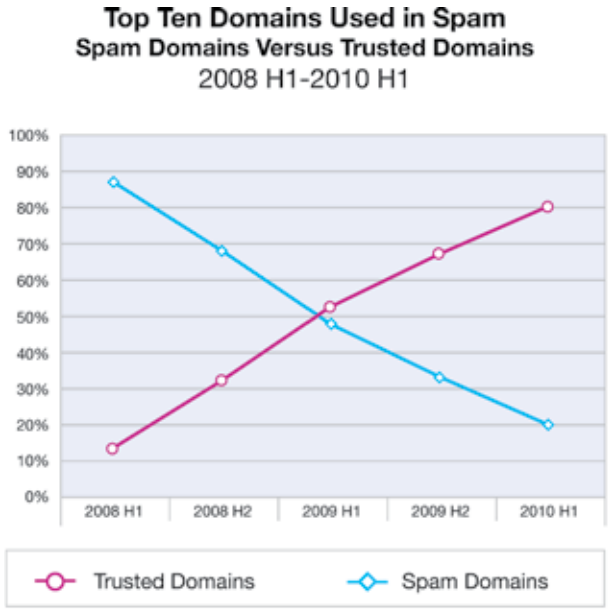


Figure 61: Top ten domains used in spam, spam domains versus trusted domains, 2008 H1-2010 H1

Percentage of random URLs per top level domains

In our earlier section we discussed how the Top Level Domains (TLDs) have moved from China to Russia possibly because of new tighter restrictions imposed by China on registering domain names. To further explore this subject we turn our attention to Top Level Domains that use random naming structures.

Regarding top level domains, there is an interesting aspect about generic top level domains (like **.com** and **.net**) versus the top level domains of a country. Spammers have many techniques they can use to make the messages look legitimate. One of these techniques is to use random domain names that are legitimate (such as ibm.com). In many cases these legitimate URLs are hidden in the HTML source code of the email. Only the one URL that links to the real spam content is visible to the user and clickable.

When analyzing country code top level domains (like **.cn**, **.ru**, and **.es**) these are not used randomly. Nearly 100 percent of these URLs really do host spam content (or redirect to Spam content automatically) if they are used in a spam message, which is different for the generic top level domains like the .com addresses. Figure 62 shows generic TLDs that most frequently use random domains (without hosting spam content). The term "random domain" means the name of the domain is randomly chosen, regardless of whether the domain really exists or not.

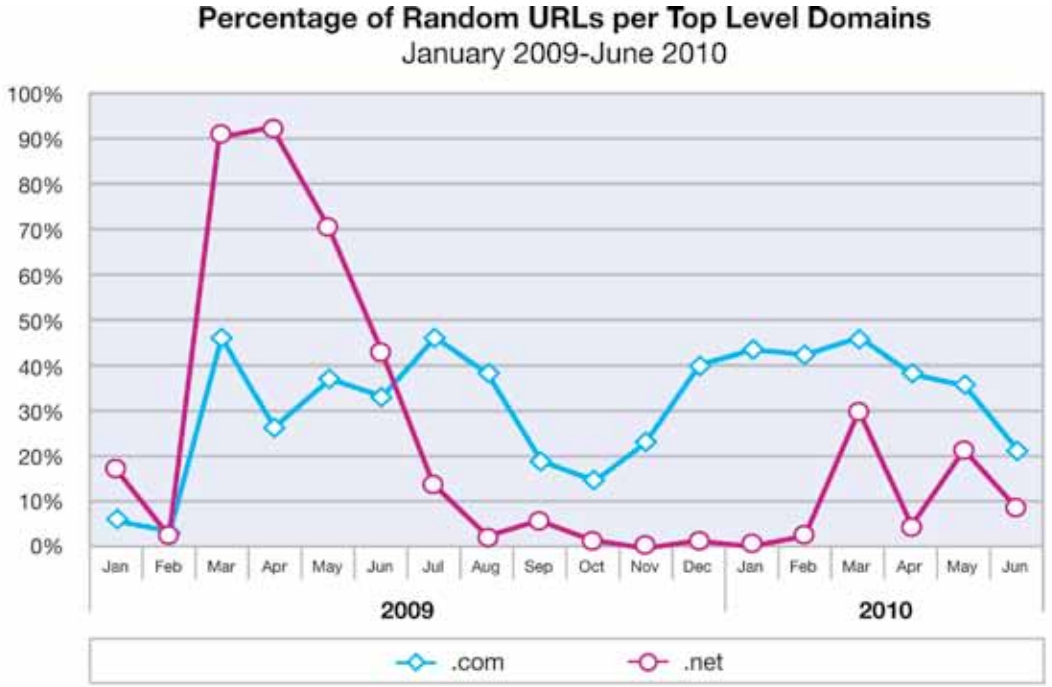


Figure 62: Percentage of random URLs per top level domains, January 2009-June 2010

As Figure 62 demonstrates, **.net** URLs found in spam emails were typically randomly generated. Fake URLs made spam more legitimate throughout the spring and summer of 2009. But since August of 2009, the use of random **.net** URLs stopped almost completely until March of 2010. Then, random .net URLs again became a popular tool for spammers. In March 2010 about 30 percent of **.net** URLs found in spam were randomly generated. In May 2010, about 20 percent of **.net** URLs found in spam were randomly generated. Random **.com** URLs were used all the time. In most cases, only 60-80 percent of them really do host spam content while 20 to 40 percent are randomly chosen. Hence, it is still a popular method for spammers to use to make their messages look legitimate.

Reputation of spam URLs: do they link back to the Internet?

Almost all spam URLs—those that host real spam content—are from newly registered domains. It is rare to find a spam URL that was previously known by crawling the Internet. Another way to look at this problem is by reputation rank, that is, to check whether spam pages link to other parts of the Internet. Figure 63 shows what percentage of spam URLs contain links to other URLs.

As Figure 63 illustrates, spammers do not tend to link to other parts of the Internet. In the first half of 2008, about 6 percent of all spam URLs contained links. Before and after that time, less than 2 percent of spam URLs linked to other parts of the Web.

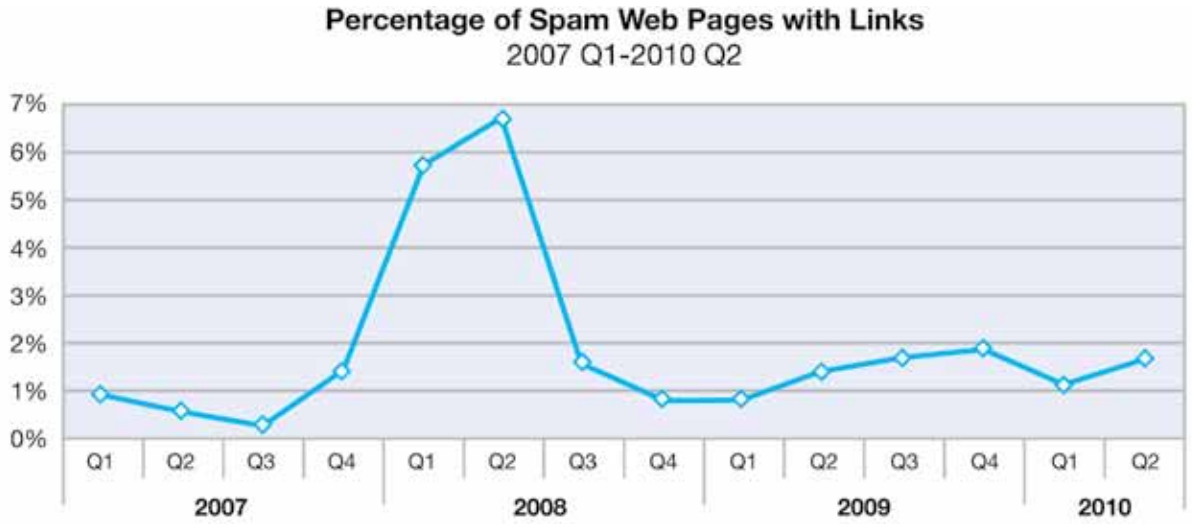


Figure 63: Percentage of spam Web pages with links, 2007 Q1-2010 Q2

Throughout 2009, however, spammers slowly increased the percentage of spam URLs with other links. At the beginning of 2010 they fell back to 1.1 percent but by mid year they reached nearly 2 percent again. Let's take a closer look at what kinds of URLs they are linking to.

Figure 64 breaks up these URLs into two categories: good categories (such as general business, shopping, software, hardware, etc.) and bad categories (like pornography, malware, anonymous proxies, and so on).

The majority of links point to good URLs. It is likely that spammers are attempting to obtain a good reputation score for their spam URLs. It is important to remember that less than 2 percent of spam URLs contain any links at all.

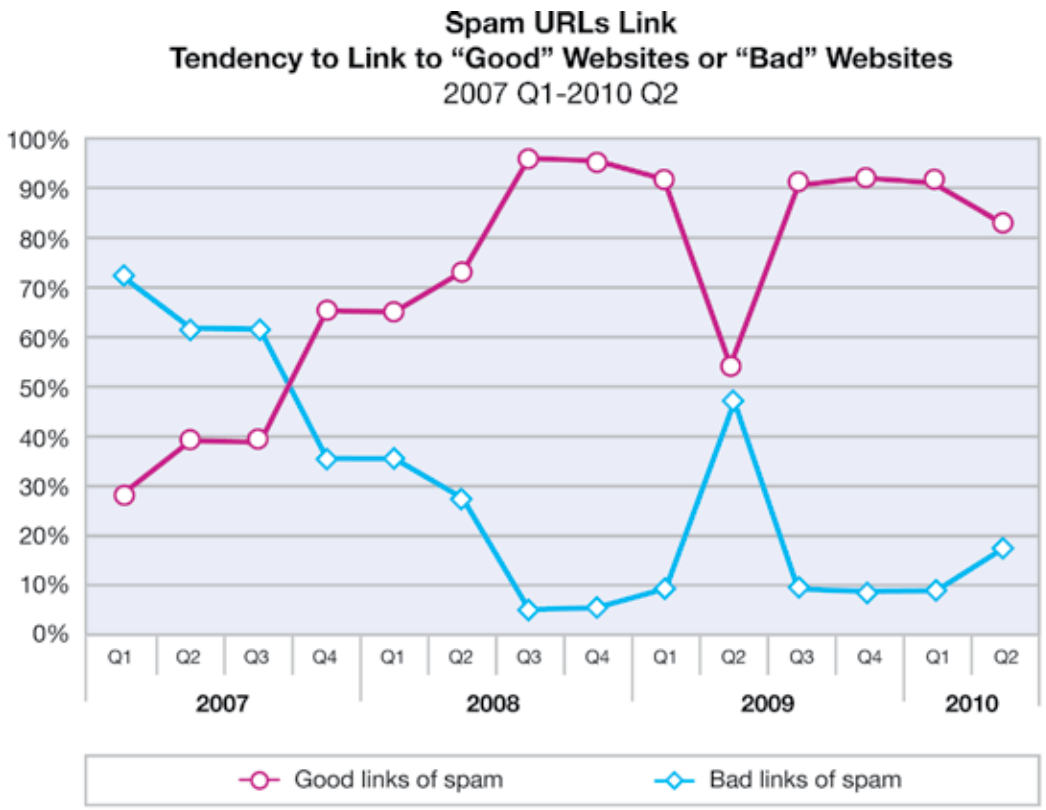


Figure 64: Spam URLs link, tendency to link to "Good" websites or "Bad" websites, 2007 Q1-2010 Q2

Types of websites linked to by spam URLs

Our analysis has concluded that most spam URLs, when they do link to the Internet, tend to link to traditionally "good" websites. However, when we break down the data into our 68 categories, the most frequented type of website is in the "bad" category-Pornography. Figure 65 shows the percentage of pornography links in comparison to other links. Notice that the single category of pornography once outpaced good websites in totality (back in the first half of 2007). During the last nine months there has been a slight tendency to place more pornography links on spam URLs. The percentage of spam URLs with pornography links increased by about one percent in each quarter.

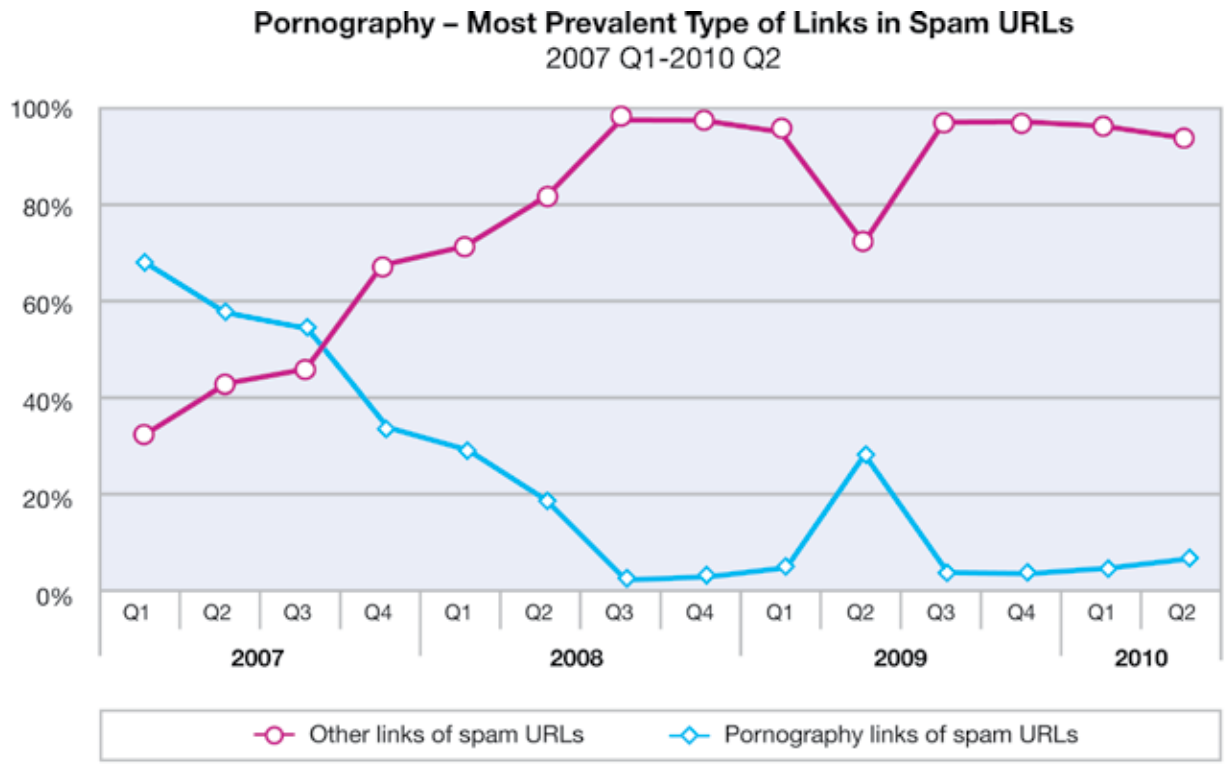


Figure 65: Pornography – most prevalent type of links in spam URLs, 2007 Q1-2010 Q2

The other major categories are good categories: general business, social networking, and shopping. At the end of 2008, social networking played a major role for the first time and accounted for more than 18 percent of all linked URLs. Although social networking links declined in the first half of 2009, they did increase slightly, reaching nearly 2 percent at the end of 2009 and then declining to 1.4 percent in the second quarter of 2010. In the first half of 2010, general business and shopping have become more attractive to spammers to give a better reputation to spam URLs.

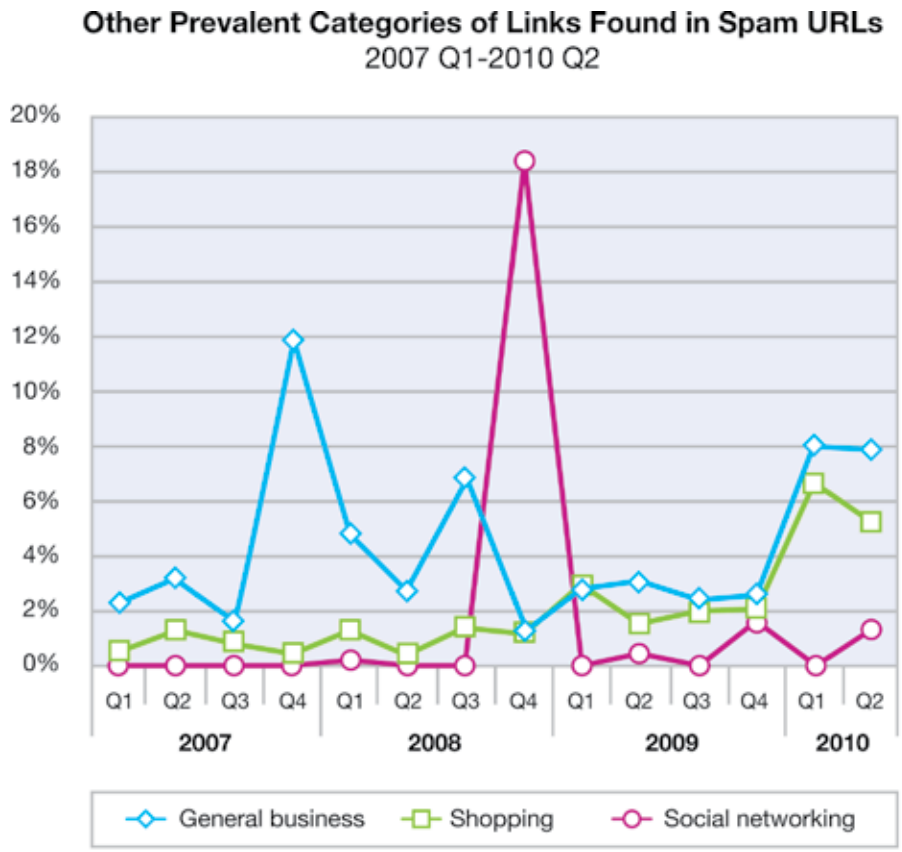


Figure 66: Other prevalent categories of links found in spam URLs, 2007 Q1-2010 Q2

Spam URLs—country of origin

Table 15 lists the origination point[2] for spam globally for the first half of 2010. Brazil, the U.S., and India account for more than one fourth of worldwide spam. The U.S. once again conquered the number one position and kept Brazil in second position. India maintained its rank in third place, Russia displaced Vietnam from fourth place and Vietnam displaced South Korea in fifth place. Germany, UK, Ukraine, and Romania are newcomers to the top ten while Poland, Turkey, China, and Colombia left the top ten spam senders in the first half of 2010 compared with 2009.

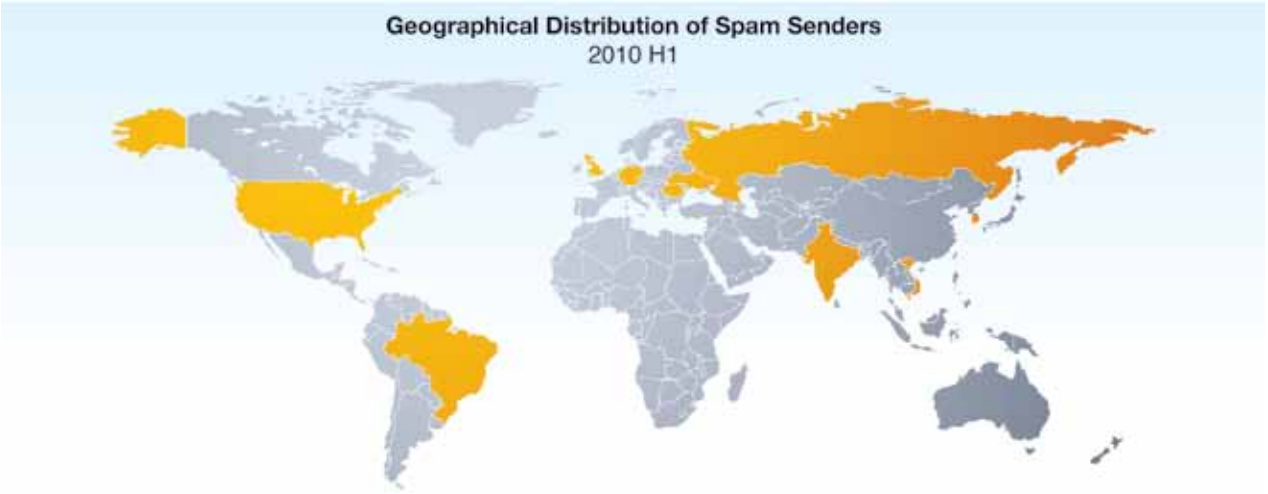


Figure 67: Geographical distribution of spam senders, 2010 H1

Country	% of Spam	Country	% of Spam
USA	9.7%	South Korea	4.1%
Brazil	8.4%	Germany	3.7%
India	8.1%	United Kingdom	3.3%
Russia	5.3%	Ukraine	3.1%
Vietnam	4.6%	Romania	3.0%

Table 15: Geographical distribution of spam senders, 2010 H1

[2] The country of origin indicates the location of the server that sent the spam email. X-Force believes that most spam email is sent by bot networks. Since bots can be controlled from anywhere, the nationality of the actual attackers behind a spam email may not be the same as the country from which the spam originated.

When looking at shorter time frames and including the previous year, some trends become visible. In 2009, Brazil had the number one position and had even extended its percentage. Except for Brazil, in the fourth quarter of 2009, Vietnam was the only country that sent out more than nine percent of all spam. On the other hand, the United States, Russia, and Turkey became much less important as spam-sending countries. But in the first half of 2010, Brazil declined significantly while the U.S. recovered. Vietnam lost ground, and India shows a continued increase for more than a year. In the second quarter of 2010, India became runner-up for the first time and needs an increase of only 0.6 percent to reach the number one position.

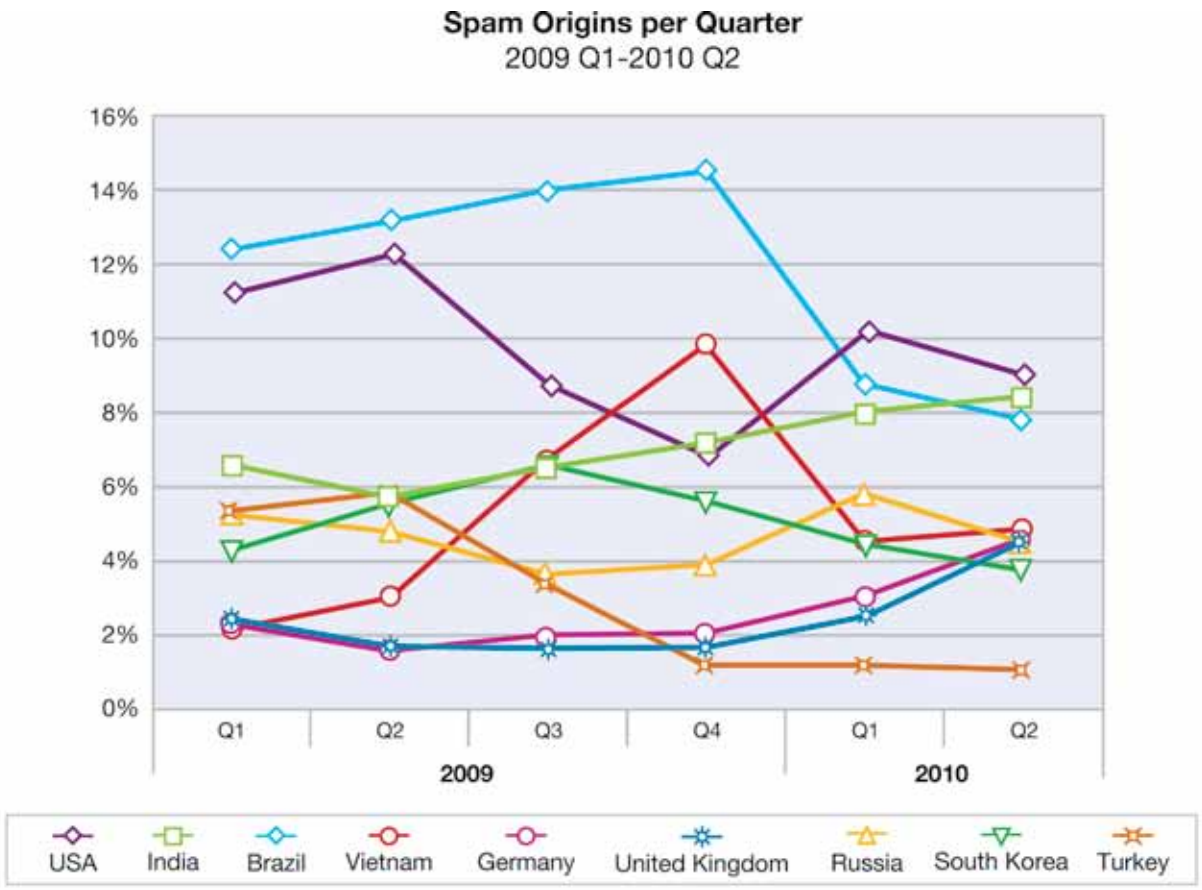


Figure 68: Spam origins per quarter, 2009 Q1-2010 Q2

Growth in BRIC countries

Brazil and India, as BRIC[3] countries, have shown rapid growth in the spam and phishing industries. In the first half of 2010, Brazil is the number one phishing sender (see more details in a later section). The other two BRIC countries, Russia and China, have not been complacent in this regard. As shown in Figure 66 on previous page, Russia's top level domain **.ru** is the favored TLD to host spam content. And, as shown in Table 16, China is the top hosting country for spam URLs. For the BRIC countries, spam and phishing are two industries experiencing growth as rapidly as many other industries in these countries.

But why Vietnam and Brazil? There may be two main conditions that need to be met to get to the top of the list of spam sending countries:

- Significant growth of the Internet-using population
- Significant number of inhabitants

Both conditions are fulfilled by Brazil and Vietnam. In Brazil, 38 percent of the 201 million inhabitants use the Internet. This number has increased by more than 1,419 percent in the last ten years.[4] In Vietnam, 27 percent of the 90 million inhabitants use the Internet. This number has increased by 12,035 percent in the last ten years.[5] These increases have lead to a large number of inexperienced people using PCs. These PCs may be less patched or protected. Or, they may be more prone to socially-engineered beguilement, making them more vulnerable to malware that could turn them into botnet drones. It is worth noting that spammers have even gained ground in well-developed countries like Germany and UK. Both increased their spam-sending "market share" to more than 4 percent in the second quarter of 2010.

Reasons for this increase might include the following:

- There are more and more inexperienced people using PCs.
- More and more viruses circumvent even well-protected systems and are successful in turning PCs into botnet drones.
- Even experienced people are not immune to the dramatic increase of vulnerabilities found in common software products.

As we reported earlier on **page 18**, the first half of 2010 has seen an incredible increase of reported vulnerabilities and the emerging economies of the BRIC countries are not excluded from this upward trend.

[3] BRIC is an acronym representing the rapidly growing economies of Brazil, Russian, India, and China.

[4] http://www.internetworldstats.com/stats15.htm

[5] http://www.internetworldstats.com/stats3.htm

Spam URLs—country of origin

Table 16 lists where the spam URLs are hosted.

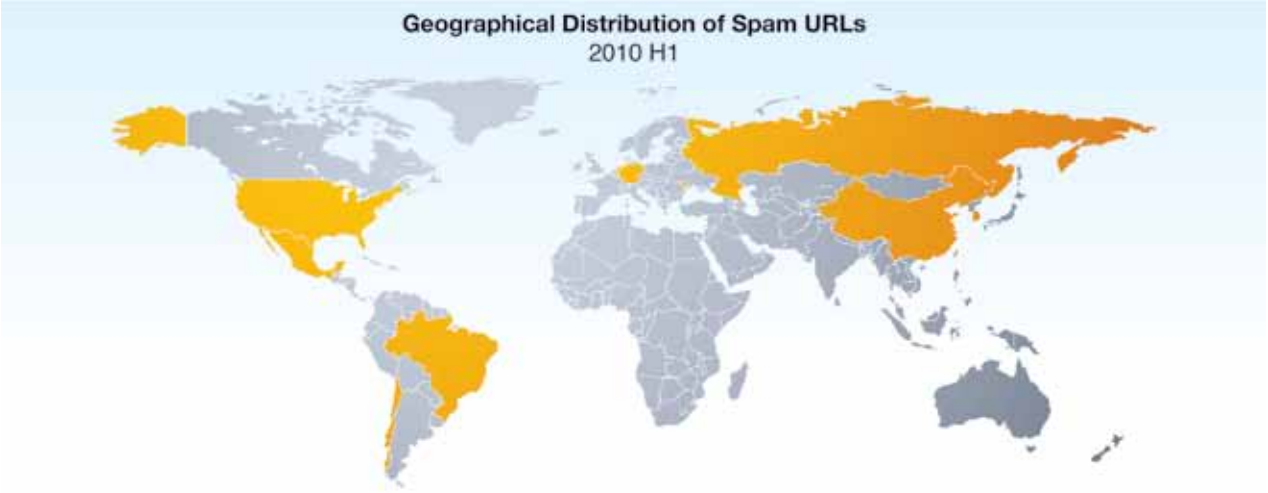


Figure 69: Geographical distribution of spam URLs, 2010 H1

Country	% of Spam	Country	% of Spam
China	37.5%	Brazil	1.9%
USA	16.6%	Mexico	1.6%
South Korea	8.9%	Netherlands	1.5%
Moldova	4.7%	Chile	1.5%
Russia	3.4%	Taiwan	1.5%

Table 16: Geographical distribution of spam URLs, 2010 H1

Spam URLs—country of origin trends

Over the last years, and until end of 2009, spam URLs hosted on servers in China dramatically increased. All other countries have stagnated or declined, particularly the United States. In the first half of 2010, the trend towards China has slowed, and China actually declined for the first time in the last two years. China still holds the number one position, hosting more than one third of all spam URLs. Some other countries have recovered, particularly the U.S., now hosting 17 percent of all spam URLs and South Korea, hosting nearly nine percent of all spam URLs. A newcomer to the top ten is Moldova, which hosts 4.7 percent of all spam URLs.

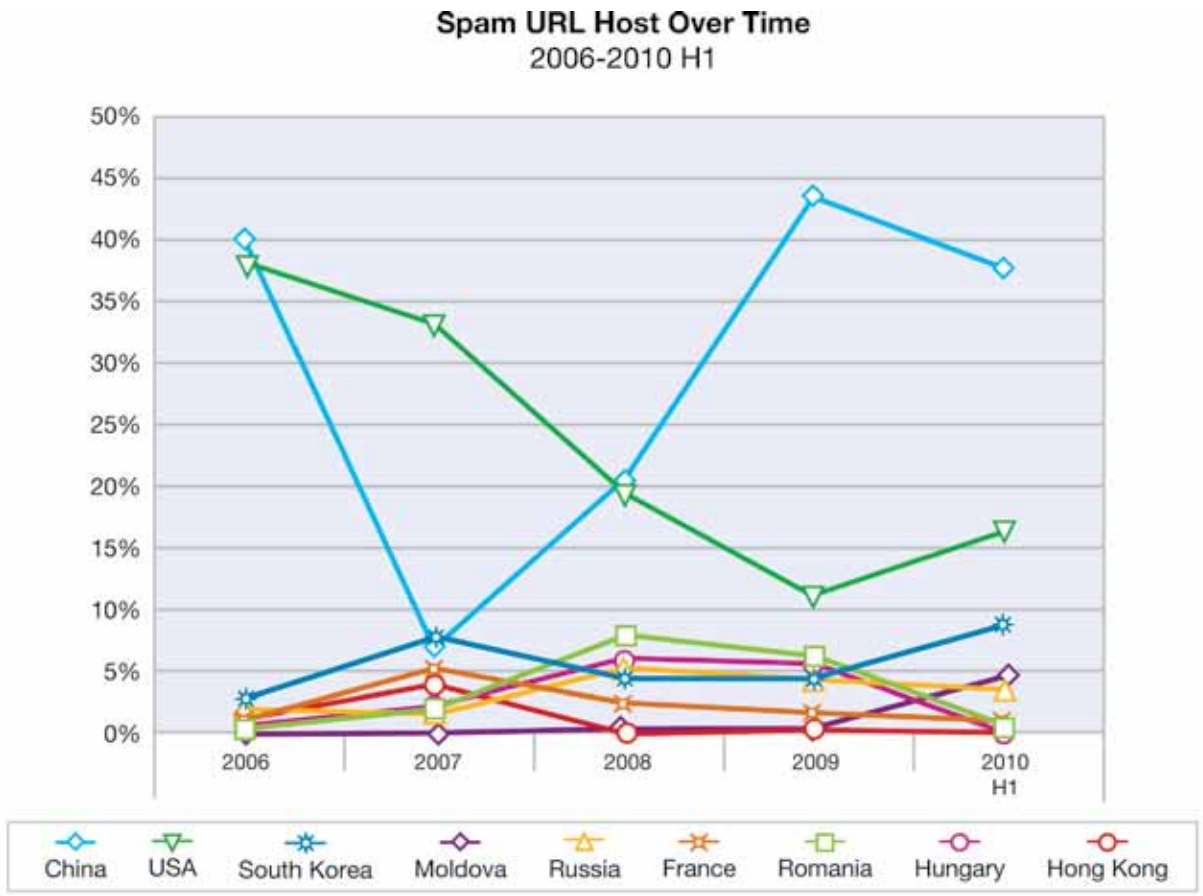


Figure 70: Spam URL host over time, 2006-2010 H1

Globalization in terms of spam

As China still dominates the spam URL hosting scene, we should look closer at these URLs, particularly in light of the massive drop of spammer's use of .cn domains. Figure 71 illustrates the distribution of top level domains used by spammers and hosted in China.

More than 60 percent of all spam domains hosted in China have the Russian top level domain **.ru**. China's own top level domain **.cn** is only a runner-up with less than 30 percent.

So, what does globalization mean in terms of spam? A typical spam is sent from a machine located in USA, India, or Brazil, contains a **.ru** URL that is hosted in China.

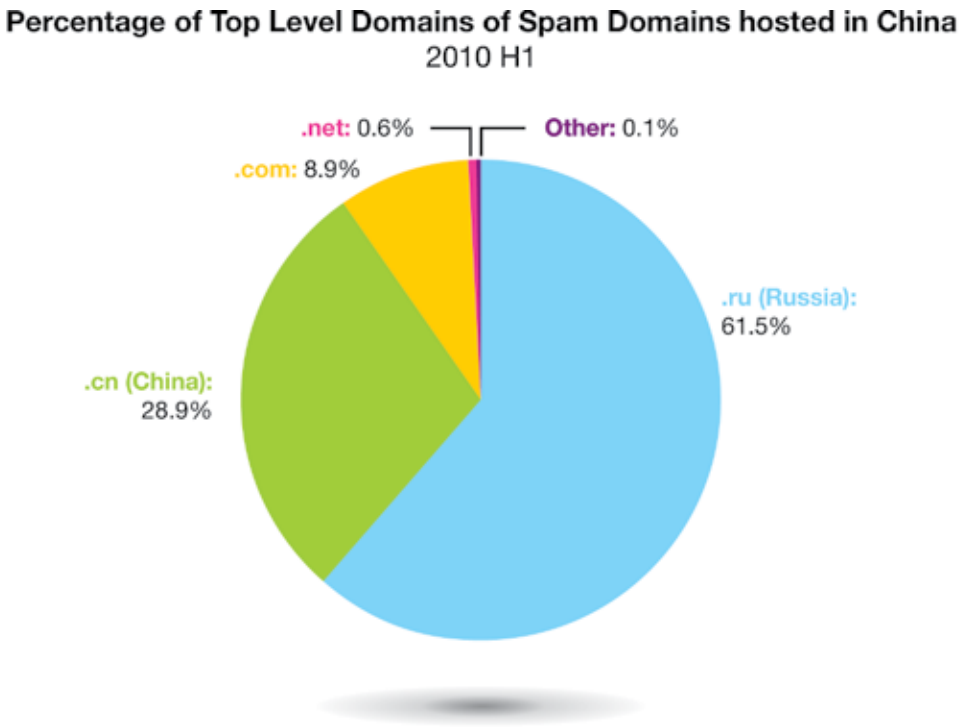


Figure 71: Percentage of top level domains of spam domains hosted in China, 2010 H1

Spam—most popular subject lines

While spam subject lines became more and more granular from 2007 to 2008, this trend is stagnating in 2010. The top ten subject lines in the first half of 2010 make up about 3.3 percent of all spam subject lines, a bit more than the 2.6 percent in 2009 and the three percent in 2008, but significantly down from the 20 percent figure recorded in 2007.

As Web 2.0 and social networks become more popular, spammers use subjects related to these topics to attract users' interest. Furthermore, the "classic" topics about medical products or replica watches are often used to attract a user's attention. Particularly medical products of Pfizer enjoy great popularity when mentioned in spam's subjects. Here spammers do it in their traditional way and play with upper and lower case, replace "o" by "0" (zero), use different percent numbers and so on. Obviously 70 percent is their favorite percentage rate, as this is the only one that reached the top 10.

Table 17 lists the most popular spam subject lines in the first half of 2010.

Subject Line	%
You have a new personal message	0.50%
Replica Watches	0.44%
RE: SALE 70% 0FF on Pfizer	0.40%
News on myspace	0.35%
Important notice: Google Apps browser support	0.35%
Important notice: Google	0.34%
Please read	0.29%
Exquisite Replica	0.23%
Watches	0.19%
Confirmation Mail	0.17%

Table 17: Most popular spam subject lines, 2010 H1

Phishing

We have already shared some exciting stories in the first section of this report on why the focus of phishing techniques is targeting different industries.

This section further explores trends relating to the following topics:

- Phishing volume as a percentage of spam
- Phishing country of origin trends, including phishing Web pages (URLs)
- Most popular subject lines and targets of phishing

Phishing volume

Throughout 2008, phishing volume was, on average, 0.5 percent of the overall spam volume. In the first half of 2009, phishing attacks decreased dramatically to only 0.1 percent of the spam volume. We thought that the criminal network behind phishing might be leaning towards other methods for identity theft other than sending out a simple email that looks like a legitimate email coming from a bank. Far from it.

Contrary to what we witnessed in the first half of 2009, phishers came back with a vengeance in the third quarter. In June, 2009, we saw a tiny uptick in volume. By August, however, the volume of phishing reached the volume seen in the most active months of 2008, and the volume seen in September completely surpassed the volume seen during any one month of 2008.

We were not the only ones who noticed—several other research organizations talked about the change. By the end of 2009, phishing slowed down to volumes similar to the end of 2008, but it was still significantly above the volume in the first half of 2009. After a slight increase in December of 2009, in the first half of 2010, phishing emails again slowed down to volumes similar to the first half of 2009.

After a drop in January and February we saw an increase in the phishing volume in March and April. In May there was another drop. This might be in relation to the apprehension of a Romanian phishing gang at the beginning of May (see http://www.h-online.com/security/news/item/Police-apprehend-Romanian-phishing-gang-997151.html). In June, the levels of March and April were reached again, but still far away from the volumes of summer of 2009. In the upcoming months we will have to wait and see if phishers will again be in full cry in summer and autumn of 2010 as they have dramatically increased their levels during these seasons for the last two years.

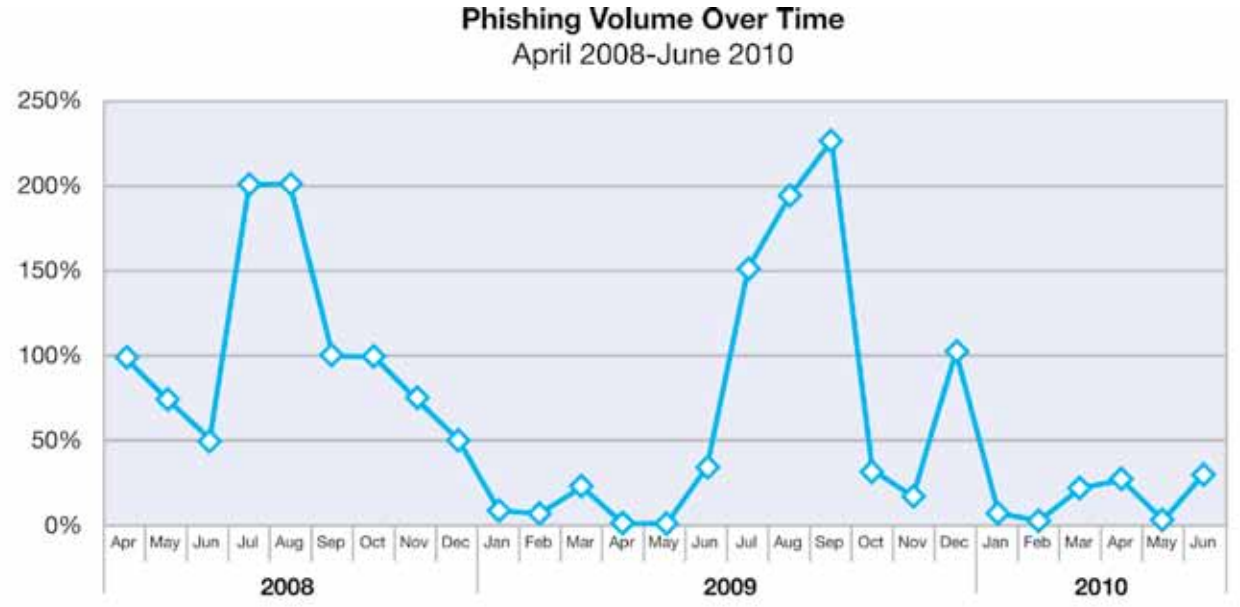


Figure 72: Phishing volume over time, April 2008-June 2010

Phishing—country of origin

Brazil is still the top sender in terms of phishing volume, while India is a runner-up in second place, and South Korea holds third place. Within the top ten, we mostly saw moves up to three ranks up or down in comparison to 2009. Only Russia fell from rank three to rank ten. Germany is new to the top 10 and Turkey disappeared. Table 18 lists the major countries of origin for phishing emails in the first half of 2010.

Figure 73: Geographical distribution of phishing senders, 2010 H1

Country	% of Spam	Country	% of Spam
Brazil	14.3%	Argentina	3.8%
India	8.2%	Chile	3.3%
South Korea	7.8%	Germany	3.1%
USA	5.6%	Poland	2.9%
Columbia	3.4%	Russia	2.6%

Table 18: Geographical distribution of phishing senders, 2010 H1

Phishing URLs—country of origin

Table 19 shows where the phishing URLs are hosted. The top ten players have not changed in comparison to 2009, and even relative ranking have only changed a little. Russia fell from rank eight to rank ten while Spain and Poland gained one rank.

Figure 74: Geographical distribution of phishing URLs, 2010 H1

Country	% of Spam	Country	% of Spam
Romania	18.8%	Canada	4.7%
USA	14.5%	Japan	4.3%
China	11.3%	Spain	3.2%
South Korea	9.8%	Poland	3.0%
United Kingdom	7.2%	Russia	2.9%

Table 19: Geographical distribution of phishing URLs, 2010 H1

Phishing—most popular subject lines

One of the biggest changes in 2008 was popular subject lines were not so popular anymore. In 2007, the most popular subject lines represented more than 40 percent of all phishing emails. In 2008, the most popular subject lines made up only 6 percent of all phishing subject lines. Thus, phishers became more granular in their targets in 2008, essentially with a greater variety of subject lines than in 2007.

In 2009, the trend reversed completely. The top ten most popular subject lines represented more than 38 percent of all phishing emails. In the first half of 2010, the top ten most popular subject lines represent about 36 percent of all phishing emails.

The text "Underreported Income" is seen four times in the top ten phishing subject lines and belongs to a phishing threat that we have seen for almost a year. It is related to a U.S. tax website. The remaining six subject lines are quite common. Most of them contain an urgent request to the user to do something. In most cases users were asked to login to their bank accounts by following a link in the email which led to a fraudulent website.

Table 20 lists the most popular phishing subject lines in the first half of 2010.

Subject Line	%
Security Alert - Verification of Your Current Details	15.75%
American Express Online Form	6.22%
important notification	1.95%
Official information	1.78%
Your Account Has Been Limited	1.73%
Notice of Underreported Income	1.70%
Underreported Income Notice	1.67%
the CP2000 notice (Underreported Income Notice)	1.67%
official "Underreported Income Notice" to taxpayer	1.66%
Final notice	1.40%

Table 20: Most popular phishing subject lines, 2010 H1

IBM Global Services
Route 100
Somers, NY 10589
U.S.A.

Produced in the United States of America
August 2010