PANDA
CLOUDINTERNETPROTECTION

Simply... **Evolution**

HOW HACKERS ARE TARGETING ENTERPRISE NETWORKS FROM THE INSIDE-OUT

PANDA CLOUD
OFFICE PROTECTION

PANDA CLOUD
EMAIL PROTECTION

PANDA CLOUD
INTERNET PROTECTION

PANDA
SECURITY

ABSTRACT

As enterprise defenses evolve, so too do the attack vectors leveraged by those seeking to bypass such controls. We are entering an era where attackers are no longer working to punch a hole in the fortress surrounding enterprise IT assets from the outside – they don't need to - they're already inside. Thanks to the meteoric rise in the importance of web-based traffic and a laundry list of vulnerabilities in web-aware applications on the desktop, end users are now being turned into enablers for enterprise attacks. Convincing a trusting employee to visit a website is all that an attacker needs to do in order to access valuable data deep within a secure network.

For far too long, enterprises have focused on expending the vast majority of available security resources on protecting enterprise servers, while ignoring the risks inherent on the thousands of desktops within that enterprise. As the security controls protecting Internet facing servers has improved, thanks to the efforts of software vendors and corporate security teams, attackers have shifted their focus to target weak desktop security and naïve end users. Defending a few hundred enterprise class servers with a team of knowledgeable system administrators is trivial compared to the challenge of locking down tens of thousands of client machines and educating the users that control them. Don't expect the challenge to get any easier; it is becoming increasingly complex as employees become more and more mobile – outside the confines of perimeter controls protecting the LAN. That fact has not eluded an increasingly well funded and organized army of attackers with no shortage of financial motivation. As attackers evolve, so too must enterprise security, by adopting solutions that enable uniform end user security across an enterprise, whether protecting a desktop at corporate headquarters, or the laptop of a road warrior browsing the web at his favorite coffee shop.

INDEX

1. Overview

Enterprise networks are commonly described as though they were a candy coated treat – hard and crunchy on the outside, soft and chewy on the inside. We build moats and impenetrable walls around the fortress that is the corporate LAN where our electronic treasures are stored. Attackers are all too aware of this fact and like the Trojan Horse from Greek mythology, they too know that it's much easier to attack such a structure from the inside, than it is to break through the many layers of security to get to the gold. However, unlike the Greeks, who leveraged a giant wooden horse filled with soldiers to get inside, hackers today already have an army on the inside waiting for marching orders. You probably know this virtual gang by their more common name - employees.

You would be hard pressed today to find an office where employees don't leverage the Internet as their single most valuable resource. Whether responding to email or browsing web pages, the Internet is a vital asset. It's also a minefield littered with security risks. Phishing sites appear from nowhere and fade away just as quickly. Social networking has turned into an art form with the attack du jour arriving as spam in your inbox or filling comments on your blog. And with the Web 2.0 era encouraging user- generated content, legitimate sites are becoming a virtual bulletin board for active content attacks and malicious binaries. Gone are the days when attackers hammered away at perimeter resources such as web and mail servers, hoping to find a neglected server without the latest patches. Finding servers with gaping vulnerabilities is getting harder thanks to increasingly security conscious network administrators and software vendors implementing secure coding practices. This is largely due to the hard lessons learned in years past. There's no need however to scour for a couple of vulnerable servers within an enterprise, when that same company has thousands of vulnerable end users.

For years, enterprises have poured the vast majority of the security budget into protecting the crown jewels – enterprise servers. After all, that's where all the important data resides. Companies have purchased firewalls and intrusion detection systems. They've hardened servers and streamlined patching cycles to ensure that externally facing resources are bullet proof. Meanwhile, end user security has largely been neglected. Sure, desktops have anti-virus engines and end users are forced to sign off on a security policy, but in a Web 2.0 world where a desktop can be compromised simply by visiting a web site, these measures have limited value. Securing a network no longer means protecting a handful of servers in the DMZ. That's the easy part. The hard part is protecting thousands of mobile end users against increasingly dynamic attacks for which patches may not exist.

2. History

The techniques employed by attackers and the countermeasures designed by security professionals have evolved over the years. This evolution has created eras defined by the changes in attack patterns used. As others have pointed out, the battle between attackers and defenders is a cat and mouse game that will never end, but the rules will continue to change. It is important to look back and learn from these past eras and peer into the crystal ball to anticipate what is yet to come.

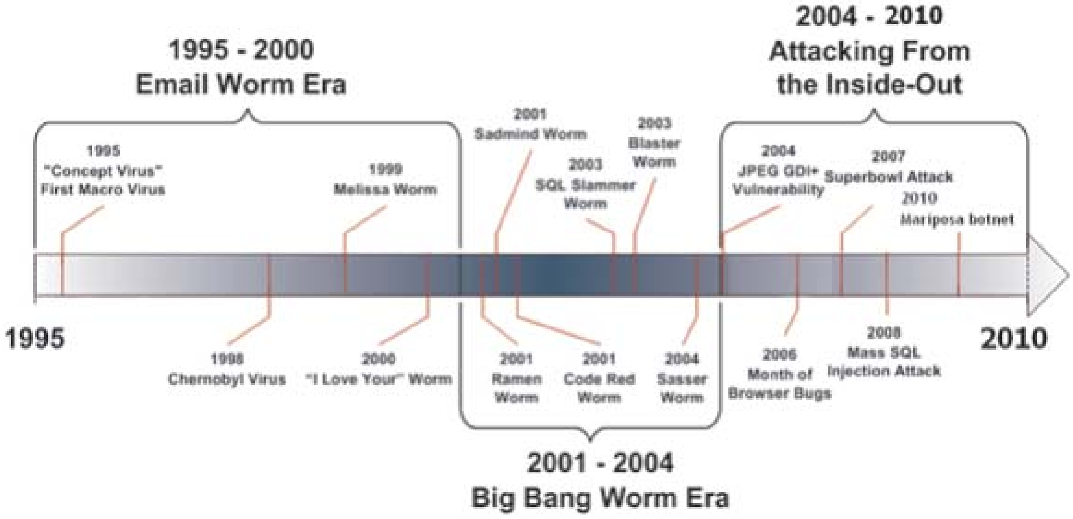


Figure 1 - An Evolution of Attack Vectors

2.1. 1995-2000 – Attacking the Client: *The Email Worm Era*

Toward the end of the last millennium, attackers latched onto the explosive popularity of email communication and combined it with social engineering to usher in the era of the email worm. Email was a perfect medium for mass attacks. Email could carry attachments that could be sent to thousands of random or targeted users with minimal effort or cost. The only challenge involved convincing end users to open the attachment and the world witnessed plenty of creativity in this regard. The "I Love You" worm, first identified on May 4, 2000, was simply a VBScript attached to an email message with the subject of "I Love You"[1]. People we so curious to see who their admirer was that hundreds of thousands of machines were infected within hours and system administrators were kept busy for days cleaning up the damage.

Attacker motivations at this time were driven primarily by fame and notoriety. In the case of the "I Love You" worm, other than overwriting certain files in an effort to continue propagation, relatively minimal damage was done and the worm was clearly not an effort to make money. David L. Smith, author of 'Melissa', another email worm, admitted during his prosecution that he had named the worm after a Florida stripper that he'd met earlier. For his efforts, Smith gained the distinction of being the first US person to be successfully prosecuted for creating a malicious program.[2] The creators of both these worms, and most unleashed during this period, may have been motivated by curiosity or ego, but not a profit motive.

2.2. 2001-2004 – Attacking the Server: *The 'Big Bang' Worm Era*

Creating a worm, which relies on human naiveté for propagation, isn't a significant challenge. Attackers were simply creating applications, whether compiled code or interpreted scripts, and emailing them to potential victims in the hope that they would be executed, thanks to some clever social engineering. Attackers elevated their skill sets during the next phase (the Big Bang' Worm Era) by instead relying on vulnerabilities in popular server applications for propagation. Fortunately for attackers, there was no shortage of gaping holes in the Internet infrastructure to take advantage of and enterprises generally assisted the process by following lengthy patch cycles.

The SQL Slammer worm, which began propagating on January 25, 2003, is a prime example of exploitation during this era. The worm spread so rapidly that it brought portions of the Internet to its knees. Thanks in part to the ability to spread via single UDP packets, in the early stages of infection, Slammer doubled its number of infected hosts every 8.5 seconds.[3] Microsoft had patched the vulnerability responsible for the success of SQL Slammer on July 24, 2002[4], however, numerous systems remained vulnerable some six months later when the worm was launched. This was true, despite the fact that researcher David Litchfield had publicly released a proof of concept exploit at the 2002 BlackHat Briefings on August 1, 2002.[5]

2.3. 2004 and Beyond – Attacking From the Inside-Out: *The 'Profit Motive' Era*

After 2004, the rapidly spreading worms slowly left the headlines of mainstream media. Did that mean the good guys had won the battle? Was the Internet suddenly safer? On the contrary, attackers have not given up the fight; they've merely altered their tactics to adjust to a changing environment. While that environment has changed in part due to tighter security surrounding Internet facing resources, it has also changed as organized crime has entered the equation. Attackers no longer launch noisy attacks that draw attention. Instead, whether exploiting vulnerabilities or conducting social engineering attacks, they do their best to stay below the radar as this maximizes the value that can be derived from a single attack.

In January 2008, we received a reminder of the new attack paradigm. Automated SQL injection attacks were conducted which resulted in malicious code being injected into at least 70,000 public websites. The servers were not the ultimate targets; they simply provided the attack platform. The code injected into the sites exploited numerous known vulnerabilities, which would install key loggers on the computers of victims, who visited the sites using vulnerable web browsers. With key loggers installed on thousands of machines, the attackers had access to confidential data such as user names, passwords and credit card numbers.

This attack took advantage of common vulnerabilities in web servers to attack those that visited the infected sites. Identifying vulnerable sites is not, however, always necessary as we learned when MySpace faced the Samy worm. Samy was a non-malicious cross site scripting (XSS) attack that automated the process of adding 'friends' to Samy Kamkar's profile.[6] The worm was launched as an experiment by the 19 year-old, but MySpace administrators were not impressed after the worm negatively impacted their network, as it rapidly spread, affecting over a million MySpace users. Samy Kamkar received probation and community service for the 'attack.'[7] This attack was made possible as MySpace, (and many Web 2.0 sites) allows users to add HTML content. Identifying and blocking undesirable content such as JavaScript, a process known as blacklisting, is used to impose security restrictions. However, it's difficult to block everything when a variety of encoding tricks can often be used to bypass restrictions – a lesson that MySpace learned the hard way.

Both of these attacks leveraged weaknesses in the server in order to attack their ultimate target – the web browser of an unsuspecting victim. This illustrates a growing trend among attackers, a trend that has emerged due to a lack of focus on protecting the browser and the promise of financial gain for those who can successfully attack it. It is also a trend that requires a shift in conventional security wisdom if we're to successfully address the threat.

3. The Players

3.1. Software Vendors

During *The Big Bang Worm Era*, no vendor, open source or commercial escaped the embarrassment of explaining to their faithful that a gaping security hole in their enterprise solutions had led to the compromise of thousands of servers. More often than not the damage hit hard and fast, arriving in the form of a fast spreading worm that attacked without discrimination. During this time, the enterprises harmed by the attacks were far from innocent victims. In every example discussed previously, patches were available for the vulnerabilities that were exploited. Known vulnerabilities were attacked but the attacks were extraordinarily successful thanks to overly long enterprise patch cycles. In some cases, it was argued that lengthy regression testing was necessary before security patches could be implemented, while at other times, those responsible simply failed to understand the risks involved in remaining unpatched. In many ways, these were the good old days – the damage was visible. Although costly, it was clear when the attack had begun and ended and it was obvious what needed to be cleaned up. As we move into an era when end users are targeted and attackers go to great lengths to fly beneath the radar, a loud and hard hitting worm seems like a luxury.

The big bang worms were directly responsible for wholesale changes in development practices. Software vendors came to the realization that without a radical change in their approach to software development, vulnerable software was inevitable. Nowhere was the impact of this new reality more obvious than at Microsoft. After embarrassing vulnerabilities in virtually all major Internet facing services exploited by worms such as Code Red, SQL Slammer and Sasser, Bill Gates issued a now famous memo detailing the need to emphasize security, availability and privacy in all software products[8]. That memo kicked of the Trustworthy Computing Initiative. Microsoft realized that brushing security on after the fact was destined for failure. Security needed to be 'baked in' from day one. That meant that everyone involved in the software development lifecycle needed to be responsible for security – not just the security team. This philosophy led to the creation of Microsoft's *Security Development Lifecycle*, architected by Michael Howard and Steve Lipner, and has resulted in a significant reduction in critical vulnerabilities in enterprise server applications.

Microsoft was also a leader with another important change during this time. Beginning in October 2003, Microsoft moved from releasing security patches at unpredictable intervals to a regular monthly patch cycle[9]. Microsoft patches are to this day, released on the second Tuesday of every month at 1pm EST. While enterprises may not know exactly what is coming down the pipeline, they can at least ensure that adequate staffing is available to handle the testing and deployment of released patches.

Another vital initiative that is now common among software vendors is the creation of security response teams. In the late 90's, if you uncovered a security vulnerability in a commercial application, you likely faced an uphill battle in your efforts to inform the affected vendor in order to obtain a security patch. At best, your challenge involved tracing down the appropriate point of contact but it was not uncommon to also face intimidation and legal challenges to keep the issue from ever seeing the light of day. Once again, vendors learned from their mistakes. They came to the realization that independent security researchers could be a valuable extension of an internal security team if resources were put in place to encourage and facilitate communication. Today, every major software vendor has a dedicated team responsible for responding to vulnerability reports and ensuring that they're addressed.

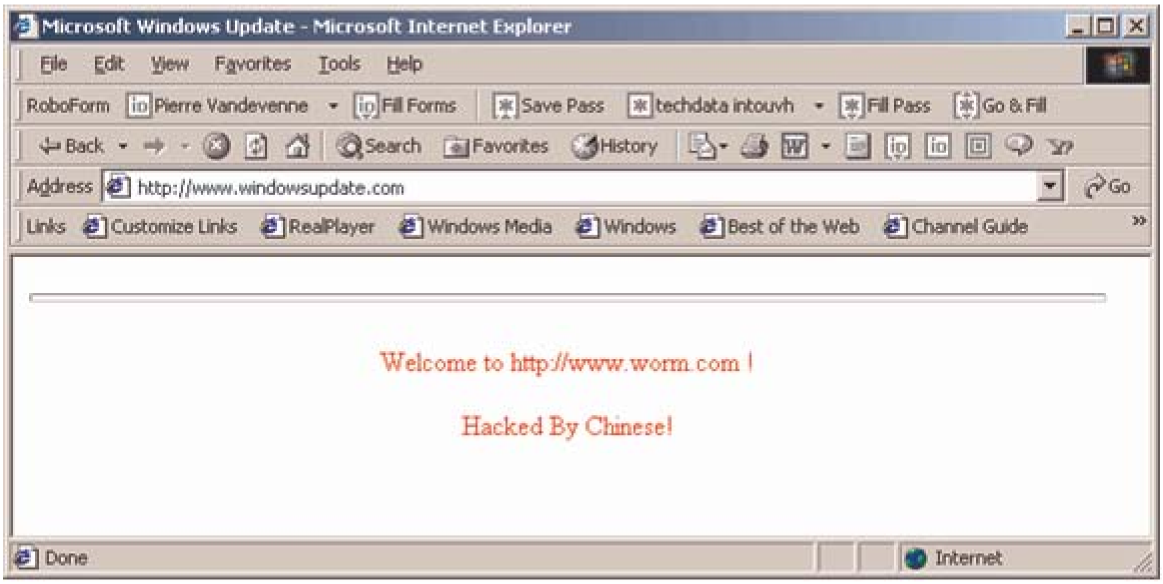
3.2. Enterprises

Enterprises have also learned from the difficult lessons of the *The Big Bang Worm Era*. While enterprise patch cycles historically spanned weeks, if not months, most corporations now realize that leaving known vulnerabilities exposed as hackers race to develop exploit code outweighs the dangers of incompatibility issues. Enterprise security teams have also become increasingly diligent when it comes to hardening servers prior to deployment and conducting both internal and third party security assessments at regular intervals.

When it comes to IT spending for security, the majority of funds have historically focused on defending servers. In the early 90's, enterprises invested heavily in firewalls to keep the bad guys out. When more granular security was required, spending shifted to network based intrusion detection and prevention systems. As we move up the protocol stack, companies have moved to web application firewalls and email security gateways. While enterprises have over the past decade, locked down Internet accessible servers, desktop access has been moving in the opposite direction. Desktop machines have gone from housing a single basic web browser capable of digesting HTML to having hundreds of applications that leverage the Web for dynamics content, updates, help files and communication. In fact, you would be hard pressed today to find a modern application that doesn't interact with web-based resources. At the same time, employees have demanded more and more open access to the web as it long ago became a mission critical resource for virtually all employees. Yet as this shift has occurred, IT spending has failed to keep pace. Beyond desktop anti-virus protection and perhaps efforts to lock down the desktop, little is done to protect end users from external threats. Attackers are all too aware of this disparity in security spending and their attacks have evolved to target vulnerable and easily accessible end user machines.

3.3. Attackers

The fame and notoriety that once inspired attackers has been replaced by a new motive – a profit motive. Many studies have been conducted over the years to try and place a dollar value on the damage that was done by malicious code outbreaks. In 2001, NewsFactor suggested that the Code Red worm had caused over $2 billion in damage and declared it the "most expensive in the history of the Internet"[10]. This cost related primarily to the man-hours required to patch and inoculate infected servers. In this case, the primary damage done by the worm was a mass-website defacement. Despite the fact that it took


Microsoft Windows Update - Microsoft Internet Explorer
File Edit View Favorites Tools Help
RoboForm Pierre Vandevenne Fill Forms Save Pass techdata intouvh Fill Pass Go & Fill
Back Search Favorites History
Address http://www.windowsupdate.com Go
Links Customize Links RealPlayer Windows Media Windows Best of the Web Channel Guide
Done
Internet

Welcome to http://www.worm.com !

Hacked By Chinese!

Figure 2 - Ironically, the Windows Update site was one of the many Code Red victims[11]

advantage of a buffer overflow that could lead to a full root compromise of a vulnerable machine, the worm was used for the electronic equivalent of graffiti. Rather than complain about the cost of cleaning up the damage, the world should have let out a collective sigh of relief that the damage was relatively minor.

Today the ‘Big Bang Worms’ are largely gone. While enterprises and software vendors can claim a certain degree of credit for improving the abysmal security, which led to such attacks, a separate factor has played an important role. Attackers today are better organized and are motivated by the opportunity to profit from their attacks. They don’t want to make the front page of the Wall Street Journal by writing a noisy worm. Once we’re aware of an attack, we can prevent it. The goal today, is to stay under the radar, exploiting technical or social vulnerabilities for as long as possible in order to make as much money as possible. In many ways, worms such as Code Red were a luxury – we knew when the damage began and concluded, and when the dust settled, the clean up was obvious. Today, attacks are much harder to identify. This is especially true when the attacks are targeted at select end user machines with minimal, if any controls to detect such attacks.

4. Attack Vectors

As attackers have shifted their focus from Internet facing servers to desktops, laptops, and mobile devices, so too have their strategies.

4.1. Social Engineering

As we're often told, security is only as strong as the weakest link in the chain. Generally, that weak link is the human element in the security equation. Social engineering, in this context, involves trying to obtain confidential information from users by tricking them into doing things that their security policy would prevent them from doing. It's the perfect combination: a carefully selected social engineering ploy convinces users to hand over their data or install a malicious program which captures information and sends it on to the fraudsters. The top 10 web exploits of 2007 all targeted high volume websites and services in order to streamline social engineering attacks. The one hurdle that an attacker must always overcome if a social engineering attack is to succeed, is that they must convince a user to access a web based resource. Historically, this was done via spam email, flooding user inboxes with URLs and hoping that a certain portion of users would click on them. Attackers are adjusting these tactics. They've realized that they can offload the challenge of generating user traffic by incorporating a popular website into the attack. Exploits such as infecting popular websites with malicious content, adding malicious user supplied content to Web 2.0 sites, or even purchasing advertising space. In most cases, the target is not the site itself; the site is simply a delivery mechanism for an attack, which targets end users.

Many times, these attacks involve no technical exploitation whatsoever. They simply convince a user to execute malicious code in order for the attack to succeed. You would think that as security technologies evolve and enterprises invest in educating end users about security threats, that the success of such attacks would diminish over time. Sadly however, social engineering attacks remain highly successful and are relatively easy for attackers to launch especially given the Web 2.0 paradigm which states "don't build a site for your users, let your users build a site for themselves". We've turned end users into web developers and without placing granular restrictions on the content that can be added, the same philosophy, which has led to the explosion in user driven content, can also be leveraged as a highly effective attack vector.

4.2. Web Browser Vulnerabilities

In July 2006, noted security researcher HD Moore launched a controversial project known as the Month of Browser Bugs.[13] Each day, throughout the course of the month, Moore released details of a new web browser vulnerability. It was controversial given the lack of coordination with affected vendors, allowing them to issue patches before vulnerability details became public. Moore did not target any particular vendor; in fact, he provided details for vulnerabilities in Internet Explorer, Mozilla, Safari, Opera and Konqueror. While his methods can be questioned, it is difficult to argue that his goal of drawing attention to the poor state of browser security was not achieved.

Web browsers have become a virtual Internet Swiss Army knife, offering functionality that goes well beyond viewing web pages. They can digest news feeds, call other applications, handle proprietary technologies and with third party plug-ins, browser capabilities are effectively limitless. As browser functionality has expanded, the underlying complexity of the applications has expanded and a flood of vulnerabilities have emerged, many of which are critical in nature and can lead to a full compromise of the host machine. Even vulnerabilities that don't result in remote code execution can have serious implications. Phishers commonly exploit browser holes to improve the effectiveness of their attacks. While browsers have built in controls to show users which resources they're accessing – address bar, status bar, SSL certificates, etc. – these controls can no longer be relied upon if vulnerabilities allow such information to be spoofed. Phishers are all too aware of this.

4.3. ActiveX Vulnerabilities

ActiveX is a proprietary Microsoft technology, which allows for reusable software components to be created. ActiveX controls can be marked as 'safe for scripting', which will permit them to be called from the Internet Explorer (IE) web browser. This technique, while not providing cross platform compatibility due to the proprietary nature of the technology, is a popular means for extending the functionality of IE. However, if such controls are vulnerable, this provides attackers with an exceptional attack vector to access a local machine. Many buffer overflows have been discovered in ActiveX controls, which can allow an attacker to execute arbitrary code on a local machine simply by convincing a victim to browse to a web page containing exploit code.

A typical Windows based computer will have hundreds, if not thousands, of ActiveX controls installed. Third party developers that might not have robust testing practices build most of these controls. Researchers eventually focused on this weakness and developed fuzzing tools to automate the process of uncovering vulnerabilities in ActiveX controls. Fuzzing tools are designed to mutate standard application input and then monitor the target application to determine if it gracefully handles the mutated input or fails and is left in a vulnerable state. The emergence of user-friendly tools such as COMRaider[14] and AxMan[15], have led to the discovery of numerous ActiveX vulnerabilities, many of which are accessible by remote attackers via IE.

4.4. File Format Vulnerabilities

File formats, like network protocols, are effectively predefined rules for communication. They define the structure of data that should be sent between computers and so long as sender and receiver adhere to the defined structure, files can be created on one machine and be interpreted on another. However, what happens when the sender strays from the format? What if bits are changed here and there? Will the receiving machine still be able to interpret the file? Will it simply discard the file, or in its effort to read the file, could a vulnerability be triggered?

File format vulnerabilities are a unique class of vulnerabilities as files are not executable code, so we don't generally consider them a threat. However, it has been discovered that malformed files can trigger vulnerabilities in an interpreting application. This creates a significant challenge for those tasked with protecting networks. While anti-virus applications generally now have signatures to detect known malicious file formats, numerous situations have emerged whereby so called 0day, or previously unknown file format vulnerabilities have been leveraged in targeted attacks. Attackers will send a malicious file to a targeted victim either by attaching it to an email message or by posting it on a website. When the file is opened by the vulnerable application that interprets it, often simply by double clicking on the file, exploitation is triggered. Blocking potentially vulnerable file types is not a realistic protection against such attacks as file format vulnerabilities have been discovered in all popular file types, including audio, video, and document formats. The attacks have created a significant challenge for Microsoft as numerous file format vulnerabilities have been discovered in file types handled by Microsoft Office applications. Given the ubiquity of applications such as Word, Excel, and PowerPoint in corporate environments, these files have proven to be an especially useful attack vector in attacks targeting individuals within an enterprise.

4.5. Web 2.0 Attacks

The term Web 2.0 does not describe a particular technology; rather it describes an evolution in the development of web resources driven by both technical and social forces. A variety of technologies such as AJAX and Rich Internet Applications (e.g. Adobe Flash and Microsoft SilverLight) are making web applications much more interactive and user friendly. This transition is gradually blurring the lines between the capabilities expected from web applications and traditional desktop applications.

While Web 2.0 technologies have not generated new attack classes, we are seeing a resurgence of traditional web application vulnerabilities in Web 2.0 applications. This is occurring for a few reasons. We frequently fail to learn from our mistakes. Enterprises often move quickly to adopt new technologies without accounting for potential security consequences. While Web 2.0 applications can and should be secure, the technologies are new for many developers who are moving to quickly learn how to develop projects using the technologies, not how best to secure them. Moreover, the technologies are new for security professionals as well, whose skill sets and tools may not yet be up to the task of uncovering traditional vulnerabilities in a new environment.

A separate issue with Web 2.0 sites is that they change the traditional dynamics of where processing occurs. In a traditional web application, all processing takes place on the web, application and database servers and the results are then sent to the web browser for display. Technologies such as RIA and AJAX can be used to produce more responsive applications largely because they offload some of the processing to the browser. In doing so, application business logic, which was not previously exposed to the end user, is now available to those willing to research the client side code. This does not need to be an issue, so long as developers recognize the consequences of such a structure. Whereas developers may have previously been able to get away with not worrying about exposing sensitive data or logic because it never left the server, they no longer have that luxury. Security through obscurity is no longer an option.

5. Trends

5.1. Attack Sophistication

The battle between attackers and defenders is an arms race, which will never end. As security vendors develop innovative technologies to identify and prevent attacks, those that perpetrate the attacks seek equally innovative approaches to defeating the new controls. Many of the most successful web attacks to date have been relatively unsophisticated, leveraging only social engineering. It is a frightening thought that we have witnessed such a high degree of attack success and much of that success has been achieved without sophisticated techniques. However, as defenses are improving and security education is reaching end users, attackers are realizing that they must raise the bar.

In recent years we've noticed a number of trends related to attack sophistication. Attackers are increasingly using 0day exploits. Such attacks rely on vulnerabilities, which have been newly discovered by the attackers themselves or purchased in underground markets. Either way, signature based defenses are useless against such attacks as signatures can only be written in response to a known attack vector.

Botnets have also emerged as the attack framework of choice. Botnets are comprised of thousands of infected hosts known as zombies, which receive instructions from command and control servers. Botnets have become a powerful force due to their resilience and flexibility. They are extremely difficult to take down in their entirety due to their decentralized nature and can be used for virtually any type of attack. When a machine is compromised and turned into a zombie, attackers are not after any of the contents on that machine. Instead, they're after CPU cycles. Cycles that can be used at will to send spam email, conduct denial of service attacks, perform click-thru fraud, or any other malicious deed that the bot herder chooses to use them for.

5.2. Economics

Attacking and controlling computer resources is now a profitable and growing business and as with any business, a number of players are involved, each trying to take a piece of the pie. Structured underground economies have evolved which link the various players. Today, it is a relatively small proportion of participants in the underground that possess the technical skills necessary to uncover exploitable vulnerabilities. Such skills are unnecessary, so long as you're willing to pay for access to them. On January 5, 2006 Microsoft raced to issue a relatively rare out of cycle patch[16]. The reason? In the days prior, it had become apparent that a wide spread file format vulnerability in Microsoft applications displaying WMF images was not only being actively exploited, but canned exploits were supposedly being sold in the underground for around $4,000[17]. The event was an early warning sign of the growing maturity of such markets. Today, cash and the right connections can provide access to exploits, credit card numbers, and attack tools. However, a more pressing concern is the growth and popularity of botnets. Rather than purchasing the products necessary to conduct their own attacks, criminals are now turning to botnet herders to gain access to services – the powerful and established botnet infrastructure that can be leveraged to unleash a variety of attacks.

5.3. Web 2.0

Beyond a technical shift, social influences are also shaping the so-called Web 2.0 revolution. Whereas Web 1.0 focused on transitioning traditional business to the web, Web 2.0 is generating entirely new business models. Web 2.0 sites focus on interconnecting users (e.g. Facebook and MySpace) and encouraging user-generated content (e.g. YouTube and Flickr). From a security perspective, this transition has greatly increased the complexity of web applications and increased the overall attack surface as users have shifted from being casual observers to active participants in building the site itself.

Web 2.0 sites are built upon the same underlying infrastructure as their predecessor sites, but add layers of complexity and push much of the application logic to the web browser. As a result, traditional web application vulnerabilities are often harder to identify. Given the interconnected nature of the web, vulnerable web servers often enable attacks against users visiting such sites. Take for example Cross Site Scripting, arguably one of the most prevalent vulnerabilities plaguing web sites today. If sites accept user-supplied input but do not properly sanitize the content, attackers may be able to force client side scripting languages such as JavaScript to execute in the victim's web browser. Once an attacker can force script execution, they can control the victim's browser. This can lead to the theft of private data, forcing unwanted actions or can even be leveraged to turn the victim's web browser into an attack tool targeting other systems on the Internal LAN.

5.4. Mobility

While warnings of attacks on mobile phones have historically been more bark than bite, that fact is changing. Mobile phones have not been attractive targets due to the plethora of target operating systems and limited functionality. While an attacker could invest time in developing an exploit for a Windows machine and instantly have a target environment consisting of millions of machines with persistent connections to the Internet, the same has not been true in the mobile market. However, as the industry solidifies around a handful of platforms (Symbian, RIM, Windows Mobile and OS X iPhone), investing time in developing exploits is becoming a more rewarding proposition. Also, mobile applications are no longer stripped down lite versions of applications with minimal functionality. Mobile Safari for example, the default browser on the iPhone, is a fully AJAX aware browser capable of digesting the same web pages delivered to desktop machines. Moreover, it shares some of the code used by its big brother on the desktop. Therefore, from an attacker's perspective, an attack that leveraged a vulnerability in AJAX functionality on a web site to infect the Safari browser, may well be able to be turned into a cross-platform exploit which doesn't discriminate against mobile users. This trend will only gain momentum as mobile platforms increase in sophistication and begin to store more and more sensitive data. Your phone isn't a phone anymore – it's a small laptop computer.

6. Challenges

Web traffic has emerged as the primary means of accessing Internet based resources due to the fact that virtually all corporate networks permit outbound access on TCP ports 80 (HTTP) and 443 (HTTPS). Applications, which may have previously leveraged alternate protocols are now likely to either exclusively use HTTP(S) or at least have the ability to revert to HTTP(S) should alternate paths be blocked by corporate firewalls. As web traffic has risen to prominence on the corporate network, so too have concerns about securing that traffic. In large disparate networks, tracking and securing that traffic comes with many unique challenges.

6.1. Uniform Protection

Web security continues to be an evolving industry and as such numerous standalone solutions have been developed by a multitude of vendors. Given the proprietary nature of such technologies and the lack of best of breed solutions from a single vendor, corporations are being forced to manage more and more software applications and hardware appliances to ensure adequate web security. As shown, in Figure 3, a single web request or response may have to traverse a half dozen separate security solutions before entering or exiting the corporate network. Combine this with the fact that large companies have multiple Internet gateways. Those gateways may also be under decentralized control or have been added to the overall infrastructure through acquisitions, so it's not surprising that a typical company is dealing with multiple web security technologies from multiple vendors, deployed at multiple locations. In such an environment, it is difficult, if not impossible, to deliver uniform protection across all locations.

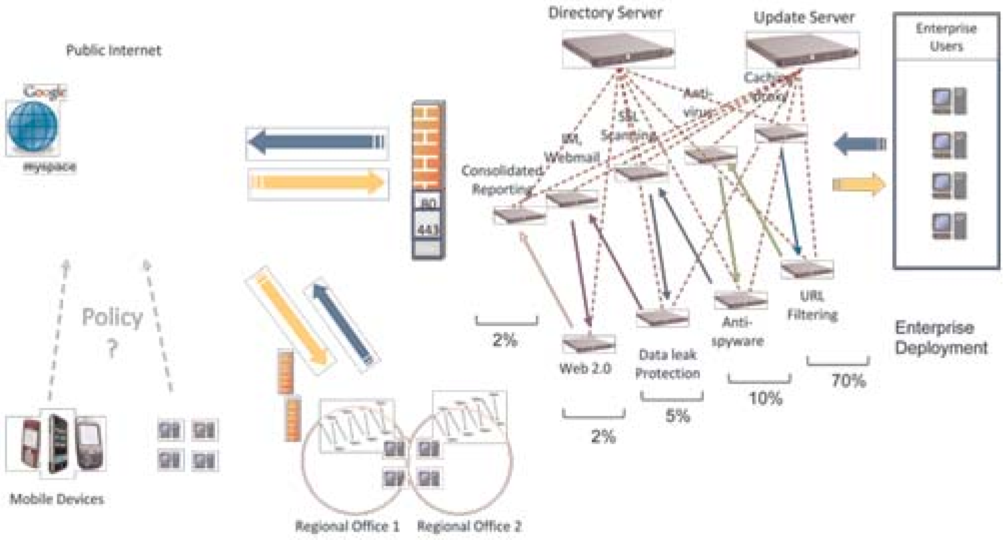


Figure 3 - Typical Web Security Infrastructure

6.2. Consolidated Reporting

Beyond the challenges of managing disparate technologies at multiple locations, a further complication is caused by a lack of consolidated reporting. It is difficult at best to manage risk when you can’t obtain a complete picture of the environment. While a variety of Security Information Management (SIM) vendors have stepped in to consolidate the reporting capabilities of individual security deployments, with each vendor adhering to their own reporting standards, SIM vendors are limited to consolidating only that data which is consistent among different vendors.

The massive log files that web traffic can generate, compound the challenge of consolidated reporting. An individual user can easily log tens of thousands of web transactions in a single day. For large enterprises even daily log files become unmanageable. These logs must then be combined in a centralized reporting system, at which point even basic data mining becomes an impossibility. What good is a reporting function, which takes ten hours to run the relational database queries required for a single report?

6.3. Consistent Policy Enforcement

Reports serve as the output of the security equation. The inputs to that equation are the policies, which define an acceptable security posture. As with reporting, when dealing with solutions from a variety of vendors at geographically dispersed locations, ensuring consistent policy enforcement may not be possible.

If different branch offices employ solutions from different vendors, enterprises are left trying to enforce a ‘best fit’ policy due to differing functionality. Decentralized security functions can also complicate matters with each branch office enforcing the policies, which they deem appropriate. For security teams that have the luxury of centralized control over policy enforcement and consistent security appliances throughout the enterprise, even then, ensuring that changes are consistently made and maintained at all locations can be a daunting task.

6.4. Data Leakage

Thanks in part to the increasing power of compliance initiatives such as HIPPA or GLBA, preventing and detecting data leakage is becoming a top priority for CISOs. Beyond compliance, the problem is exacerbated by the ease with which sensitive data can now be transferred beyond the borders of the enterprise LAN. Whether intentional or unintentional, it takes minimal effort to transfer data by way of the web, email, IM or P2P clients. Enterprises need consolidated solutions that can enforce standardized Data Leakage Protection (DLP) policies across technologies, at all web gateways.

6.5. Unknown Threats

It is often said that you can’t prevent what you don’t already know about. That’s not entirely true, but any time that attackers have the upper hand with proprietary attack knowledge, it

certainly changes the rules of the game. Many of today's security technologies are purely signature based – in which case, the attack must be known in order to build a signature to detect it. Today, with 0day vulnerabilities being discovered and traded in the underground long before patches and signatures are created and deployed, security controls must move beyond signatures to understand network behaviors to differentiate between 'good' and 'bad' traffic. For example, when a user views a site with a self-signed SSL certificate, that is hosted in China and is pulling images from a legitimate 3rd party website elsewhere, despite evidence of a known attack, adequate red flags exist to caution the end user or block the communication altogether. In a world where new blended threats emerge daily, signature based solutions provide limited protection.

Unknown threats can also be combated by employing the 'principle of least privilege'– a long-standing rule in the security world which is often cast aside to appease end users. Is it unreasonable for a user to request access to their favorite instant messaging (IM) client? Probably not. It may have a legitimate business purpose and even if it doesn't, keeping employees happy can't simply be ignored. However, does that additional piece of software increase the enterprise security risk? Definitely. Every additional piece of technology added to a corporate network increases the likelihood of inheriting vulnerable code. A balance must of course be reached. One way to reach such a compromise is to allow access to additional applications but to control the functionality. That same IM client can be deployed but controlled. Enterprises need solutions, for example, that allow text based IM conversations but restrict file downloads to protect against social engineering attacks, which convince users to download and execute files. While any solution will be a compromise, granular control over web applications and network aware desktop applications will provide options that go beyond a simple allow/disallow decision.

6.6. Cost

As the web increases in complexity and new attack vectors emerge so too do the costs associated with implementing robust security. Gone are the days when a firewall and some desktop anti-virus software was sufficient protection. Today, companies are purchasing appliances to handle in-line anti-virus scanning, DLP, URL filtering, SSL inspection, reporting, etc. Each new appliance requires a significant capital outlay, which must be duplicated at each web gateway, and this cost does not take into account the manpower required to deploy and maintain solutions. Enterprises are looking to turn fixed and often unexpected capital expenditures into predictable variable costs.

7. Solutions

Securing and controlling user generated web traffic is no longer an option in the enterprise, it's a necessity. The question therefore, is how best to approach this challenge in a way that provides the best possible control, in a cost effective manner.

7.1. Client Defenses

Client based defenses refer to software solutions which reside directly on the client machine and come in many forms. For example, you would be unlikely in this day and age to find an enterprise desktop installation that didn't have an anti-virus engine installed. Beyond this, anti-spyware and even host based firewalls and intrusion detection systems (HIPS) are becoming increasingly popular, with certain functionality being embedded directly into the operating system and not requiring a separate third party installation. However, much of this protection is designed to protect the machine itself from a direct external attack. It is not designed to detect or prevent web based attacks that are triggered by user actions. Consider for example, a cross-site scripting (XSS) attack which sends user authentication credentials to an attacker controlled server when the victim simply browses a vulnerable website. In this situation, the vulnerability existed on the web server, not the client machine, yet it was the user who was impacted. None of the aforementioned client side security technologies would be capable of preventing such an attack.

Two immediate disadvantages to any client side defenses are the overhead of installing and maintaining such solutions and the lack of control to enforce their proper use. In large corporations, manually installing client applications is generally not an option due to the cost and effort involved. This is especially true when remote employees are involved who may never physically come to the office in the first place. Beyond this, determined users can always bypass client side solutions. Employees may do so not to be malicious but simply out of frustration as they are trying to complete work, which is prevented, for example, by an overly aggressive anti-virus engine.

7.2. Web Security Gateways

The Web Security Gateway (WSG) market today, consists primarily of appliance vendors, which build solutions on top of high performance proxies, designed to inspect inbound/outbound web traffic. WSG appliances reside at one or more Internet gateways throughout the enterprise and have an inherent advantage over client defenses in that end users cannot disable them. While generally considered as a complimentary control rather than a replacement for client defenses such as desktop AV engines on HIPS, WSGs offer enterprises enhanced control over web traffic and can therefore greatly enhance an enterprise's security posture in the face of an increasing volume of web based threats.

WSGs are not however without their downsides. They add yet another appliance to the mix, which must be deployed and managed. A separate appliance must be deployed at each individual web gateway, which not only increases overall costs, but for large enterprises, can lead to a lack of consolidated reporting. Even if log consolidation is possible, the sheer volume of log entries generally makes true data mining impossible.

7.3. Cloud Security

Cloud based or SaaS deployments for web browser security offer some unique advantages. Industries such as CRM (e.g. Salesforce) and ERP (e.g. NetSuite) have begun to move toward cloud computing to enjoy such benefits such as ease of use and lower TCO. SaaS solutions are leveraging these same advantages for client-side web security.

7.3.1. Uniform Protection

The appliances detailed in Figure 3 are typically provided by a variety of vendors. No single appliance vendor is a leader in all areas of the WSG market. Therefore, enterprises are faced with a difficult choice – standardize on fewer vendors but accept second tier functionality, or go for 'best of breed' solutions in each category and deal with incompatibilities among solutions. In the end, enterprises are left choosing between the lesser of two evils. SaaS solutions are now emerging that offer broad functionality, addressing all functional areas highlighted in Figure 3, managed from a single, web interface.

7.3.2. Consolidated Reporting

With all traffic traversing through the same managed infrastructure, the SaaS solution provider can handle the headache of log consolidation. From an end user perspective, logs from all enterprise gateways and road warriors can be viewed and mined via a web application front end.

7.3.3. Consistent Policy Enforcement

SaaS solutions do not suffer from the same headaches of pushing policies to a multitude of devices responsible for policy enforcement. All enterprise web traffic is re-directed through geographically dispersed gateways, which draw from consolidated, enterprise defined policies. This way, uniform protection can be applied to all web traffic regardless of origin. When changes are made to security policies, they are made in one location, via a web-based interface. Therefore, no proprietary software is required to manage the system.

7.3.4. Data Leakage

Data leakage poses both confidentiality and regulatory risk. While appliance based DLP solutions can provide a reasonable defense against such risks, they suffer from the challenges of uniform policy enforcement across all gateways and report consolidation. These challenges are particularly compelling in the DLP space where regulatory non-compliance can have significant consequences. With SasS based solutions, where all data flows through a centralized infrastructure, these challenges can be addressed.

7.3.5. Cost

By moving the technology from the enterprise LAN into the cloud, a significant capital outlay is shifted to a per user variable expense. Beyond moving from fixed to variable costs, cloud based solutions can offer a lower total cost of ownership, due in large part to the elimination of manpower required to deploy and maintain purchased appliances.

8. Conclusion

As attackers shift their focus, so too must those tasked with defending network assets. The enterprise network can no longer be viewed as a solitary, impenetrable fortress. It has evolved into a distributed infrastructure, not just in terms of branch offices, but also with an increasingly mobile workforce entrusted with key digital assets. The goal can no longer simply be keeping the bad guys out. Enterprises need the ability to manage end user web access in order to prevent attackers from recruiting unwitting employees to do their dirty work for them. This must be accomplished without imposing overly restrictive controls on increasingly empowered employees in order to strike a delicate balance between a secure and productive workforce.

As all Internet based traffic migrates to ports 80 (HTTP) and 443 (HTTPS), this open door to the enterprise can no longer be left unguarded. The challenge facing enterprises involves identifying cost effective solutions that will address the multitude of threats surrounding end user web traffic. While client defenses such as desktop anti-virus/anti-spyware and HIPS are likely to remain important enterprise security controls, they will never be adequate solutions to provide complete protection. It is clear that enterprises must also be able to manage all enterprise web traffic. This includes not only traffic entering and leaving the corporate LAN but also the traffic to and from the laptops of road warriors and any mobile device with access to corporate assets.

While the WSG industry has emerged to meet this challenge, appliance based solutions will continue to face the limitations imposed by an increasingly distributed workforce. Uniform protection, consolidated reporting and consistent policy enforcement can be challenging when managing multiple gateways. Beyond this, it is simply not possible to protect remote users that access the web directly, not through the corporate VPN. As corporate communication continues to move toward web-based solutions, the importance of managing web traffic moves beyond security to include the additional challenges of DLP and bandwidth control. Managing this traffic has become a challenging and costly endeavor. While appliance based solutions have paved the way in the WSG market, enterprises are realizing the advantages of moving to a sole solution provider with uniform protection and control for all web traffic, regardless of location or device. Just as in-the-cloud or SaaS based offerings are emerging as market leaders in other industries, it is clear that the advantages of such a model can now also be realized in the WSG market as well.

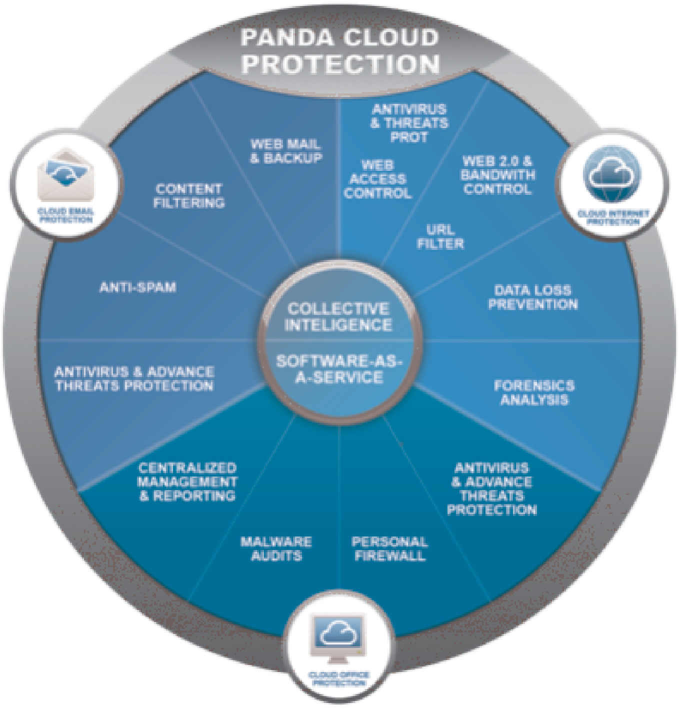
9. Panda Cloud Protection suite

Panda Cloud Internet Protection is part of the Panda Cloud Protection suite which is a complete SaaS security solution that protects all the main threat entry points: endpoint, email and Web traffic, against malware, spam, phishing, cross-site scripting and other advanced Web 2.0 attacks, through a light, secure and simple solution.

As the security suite is based in the cloud it offers maximum protection, while optimizing costs and productivity. Deployment takes just minutes, and day-to-day management is handled easily using Panda's unique and intuitive Cloud Management Console.

The Panda Cloud Protection suite harnesses the power of Collective Intelligence. Panda's cloud-based Collective Intelligence leverages 21 terabytes of knowledge and experience drawn directly from millions of users to deliver comprehensive, instantaneous, non-intrusive real-world protection against known and unknown malware to all users.

Panda Cloud Protection leverages the power of the cloud to not only provide up-to-the-minute protection against known and unknown threats but also to streamline the delivery of that protection through the anytime, anywhere power of the Cloud Management Console.



10. References

1. http://www.pandasecurity.com/homeusers/secutiry-info/about-malware/encyclopedia/owerview.aspx?idvirus=28492

2. http://news.zdnet.com/2100-9595_22-517177.html

3. http://www.wired.com/wired/archive/11.07/slammer.html

4. http://www.microsoft.com/technet/security/bulletin/Ms02-039.mspx

5. http://www.blackhat.com/presentations/bh-asia-02-litchfield.pdf

6. http://blogoscoped.com/archive/2005-10-18-n81.html

7. http://it.slashdot.org/article.pl?sid=07/02/04/1439222

8. http://news.cnet.com/2009-1001-817210.html

9. http:// news.cnet.com/Microsoft-releases-monthly-security-fixes/2100-7355_3-5091835.html

10. http://www.newsfactor.com/perl/story/12668.html

11. http://www.theregister.co.uk/2001/07/20/code_red_bug_hits_microsoft

13. http://blog.metasploit.com/2006/07/month-of-browser-bugs.html

14. http://labs.idefense.com/software/fuzzing.php#more_comraider

15. http://www.metasploit.com/users/hdm/tools/axman/

16. http://www.microsoft.com/technet/security/Bulletin/ms06-001.mspx

17. http://www.eweek.com/c/a/Security/Researcher/-WMF-Exploit-Sold-Underground-for-4000/

18. http://www.zdnet.co.uk/news/security-threats/2010/05/25/botnets-price-for-hourly-hire-on-par-with-cost-of-two-pints-40089028/

PANDA SECURITY

EUROPE
Ronda de Poniente, 17
28760 Tres Cantos. Madrid. SPAIN

Phone: +34 91 806 37 00

USA
230 N. Maryland, Suite 303
P.O. Box 10578. Glendale, CA 91209 - USA

Phone: +1 (818) 5436 901

www.pandasecurity.com

PANDA
SECURITY