Secure Coding.
Practical steps to defend your web apps.

PROTECTING USERS: THE IMPORTANCE OF DEFENDING PUBLIC SITES

GIAC (GWAPT) Gold Certification

Author: Kristen Sullivan, Kristen.Sullivan@ky.gov
Advisor: Rodney Caudle

Accepted: June 21, 2010

Abstract

One of the biggest struggles in the field of application security is trying to convince customers and clients to pay attention to the security of public facing sites that do not transmit sensitive data. In the current state of "Cyberland", we have improved at protecting our infrastructure and creating robust firewalls. We have even started to improve the protection of sites transmitting sensitive data online with the implementation of solid input validation and other related methods. However; it's becoming more and more evident that our weaknesses are our public facing sites which transmit no sensitive data and could serve as gateways to sensitive information and networks. Government agencies are tasked with providing services to constituents. To fulfill their mission these agencies must maintain the trust of the general public. Cross-Site Scripting is one of the most prevalent and dangerous vulnerabilities to web applications, especially government applications that hold the information and data of each and every citizen. Wisely investing security dollars on applications not transmitting sensitive data will help resolve these weak points in the enterprise.

1. Introduction

In the application security industry, one of the hardest elements to communicate to customers is the need for building secure web applications even if those applications transmit minimally sensitive data. The purpose of this document is to provide a valid case for why all applications should follow a minimum standard for secure coding practices. Many assume the only applications requiring protection are those which store sensitive or confidential data, but that is a grievous misjudgment. Additionally, with tight budgets and limited security resources, it is hard to justify reasons for securing public facing sites only offering open record information. The main cause of this is a lack of understanding the risk associated.

The approach to this paper will be from a governmental perspective. State and local governments all over the country are suffering from financial difficulties due to the economic crisis. News articles about government furloughs, layoffs, cut backs in services, etc. are popping up everywhere. It is important to note that when the economy starts to recover, government lags behind by at least a couple of years, so these hardships are long from being over. When the search parameter "city budget cuts" is entered into a Google search, over 8 million records can be found and likewise if "state budget cuts" is entered; over 11 million records pop up. Put simply, cuts in local and state government security budgets create new challenges in justifying the need to implement security in all of the applications the governmental bodies have to maintain. To make matters worse, awareness and training dollars for developers learning to write secure code are even less likely to be obtained.

With such tight budget constraints, government organizations are making a concerted effort to offer more and more services online since easy-to-use, self-service applications are often money-saving solutions in the long term. The new services make positive headlines and empower citizens to obtain services on demand. The latest buzz is often known as "Open Government" or "Government Transparency." For example, Government Computer News (GCN) released an issue in July, 2010 with the following cover story: "Great Gov Web Apps, 10 Examples of Innovative, Practical Web-Based Tools That Deliver Effective Services, Collaborative Environments and Data Transparency" (Government Computer News, 2010). Unfortunately, the news that

security typically generates is only negative media. There are no news articles that talk about how robust defenses saved the identities of the taxpayers. Consequently in hard times, it's much more difficult to sell preventative measures to executives. However the need is as important as ever, perhaps even more so. Since governments serve as data custodians and store tremendously large amounts of personal and corporate information, the necessity to protect the constituents is paramount. The consequences of government applications being the root cause of a mass number of constituents being attacked by cyber criminals are serious. Government entities are being held accountable for breaches like never before. According to a survey from the Pew Research Center, 80 percent of Americans say they have little trust in government (Pew Research Center for the People & the Press, 2010). This anti-government feeling has fueled the creation of the Tea Party movement, which is supported by 19 percent of Americans according to a recent poll from CBS News (Condon, 2010). Take for instance when WikiLeaks broke the story of the Virginia Prescription Monitoring Program breach. Hackers deleted more than 8 million records and defaced the site with a note stating that for $10 million dollars, they would return the records (Harlow, 2009). It was a month before any of the victims were notified (Harlow, 2009). Incidents like this fueled tighter legislation such as the HIPAA and FTC Red Flags Rule.

To this end, the U.S. Department of Health and Human Services has a list of breaches affecting the private health information of 500 or more individuals posted on their website (U.S. Department of Health and Human Services, n.d.). Many of those listed are government agencies or government funded agencies. This list is a result of section 13402(e)(4) of the HITECH Act (U.S. Department of Health and Human Services, n.d.).

The Privacy Rights Clearinghouse allows a user to search the database with various parameters (The Privacy Rights Clearinghouse, n.d.). If all types of breaches are selected, both government and education options are checked and all years selected (2005-2010), 2011 data breach records are returned. Consider that these breaches are merely those made public. Additionally, a recent a study of over 500 data breach forensic investigations conducted by Verizon Business Security Solutions, found that the publicly

reported breaches are "just the tip of iceberg" with less than 5% of the more than 500 cases covered in the Verizon study involving some form of disclosure (Claburn, 2008).

It is also important to note that these lists and studies are more geared towards more obvious losses of data. This type of loss is more tangible and easily reported than an attack on users of a public facing government internet site. Although one type of compromise is very different from the other, both are of equal importance. The current lack of trust in government entities will be exacerbated until government entities ramp up the protection of citizens.

2. Background

The intelligent criminal uses conventional everyday information to begin to attack their victim. A great point made by Ira Winkler is that the criminal doesn't care how they get the information, they're just trying to steal it however they can. Their biggest concern is how to execute the attack easily and without getting caught. (Winkler, 2005).

Those in the security industry know a key factor about attacks: once the good guys have found a way to block an attack, the attacker finds a new way to get in. According to Senior vice president of marketing at ArcSight, Reed Henry, "Whitelisting, software patching and other preventive approaches are best practices and must continue when it comes to protecting against attacks, but they will always be one or two steps behind the cyber criminals" (Raywood, 2010, para. 2).

It is a fact that network layers are more robust than they used to be. It is also a fact that web applications holding sensitive data are becoming more robust and standards are increasingly rigorous each year (i.e. PCI, HIPPA, IRS Publication 1075, FISMA, etc.). More and more legislation is being passed, not just in the United States, but across the world mandating regulations to secure web applications. With that in mind, the security experts have to pay special attention to the weakest links.

2.1 Weakest Links

On the whole, the weakest link tends to be the public facing applications which transmit no sensitive data such as social security numbers, credit card data, etc. These public facing applications are gateways for hackers to initiate massive malicious attacks on high traffic sites. Unfortunately, many people are unaware of this threat. Part of this lack of awareness can be explained by the application owner missing the value for a hacker wanting to exploit the target, even when he/she has consulted with a security expert and has been informed of the risks. Sometimes understanding the enemy and being able to think like him/her is a key in understanding what is important to protect (Winkler, 2005). Most government executives and financial decision makers have little desire to spend their already limited security budgets on applications which (in their opinion) hold no risk to their assets or their customers' personal data. The figure below from Gartner illustrates the gap between web application security attacks and the percentage of budget dollars allocated to application security (Jackson Higgins, 2010). Furthermore, Gartner predicts that the average percentage of overall IT budgets spent on security will decrease by 5 percent this year after a 6 percent drop the previous year (Jackson Higgins, 2010, para. 1).

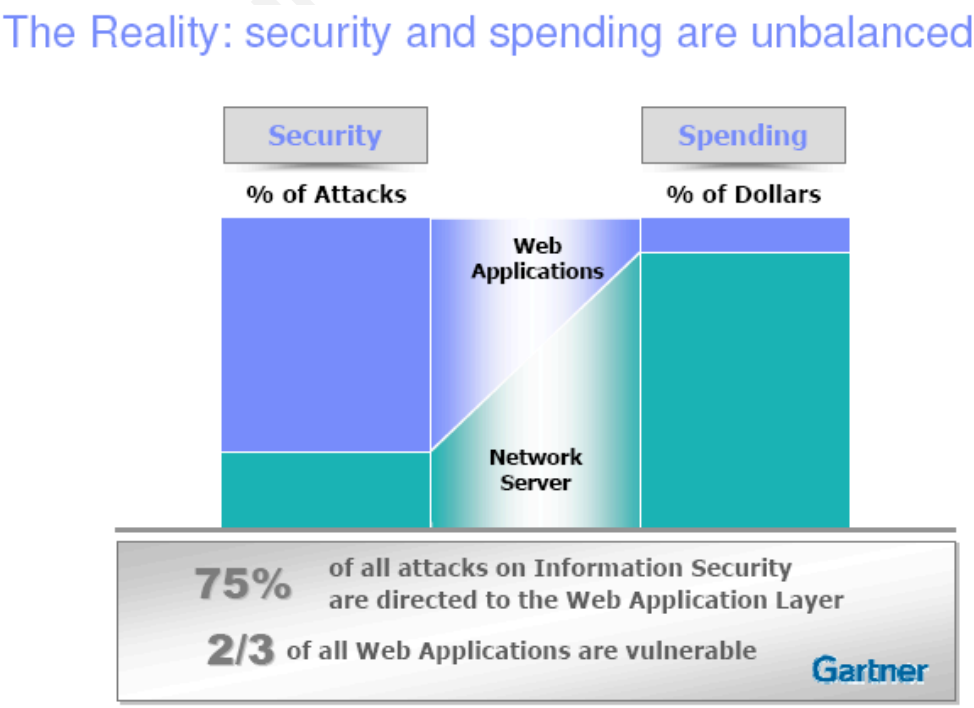

Figure 1: The Reality: Security and Spending are Unbalanced

These decision makers may not have any idea what hackers stand to gain from attacking their users via these types of public web applications. They might not even know that the public facing sites could also serve as pivot points for attackers to drill much deeper down into protected, internal assets. This lack of knowledge combined with robust tools which can automate simple attacks are both keys to the attacker's success. Hackers do not have to be extremely skilled computer scientists to execute attacks as there are so many resources and tools online that make their work relatively easy. An attack on users of a public site is akin to burglars finding an unlocked door that nobody ever guards. There isn't even a camera to catch them. Hackers are least likely to get caught executing an attack that no one will notice.

This is especially important in today's world where many servers are shared, hosting many different types of applications which serve different types of data. This makes the attack surface more complicated. One application may have great protection and may transmit sensitive data, but if it shares a server with another vulnerable application, it may still be vulnerable as well. If an attacker can break into one of the applications one a server, they may very well be able to attack everything else on a server. This is another reason it is so important to protect all web applications despite the level of sensitive data they transmit.

2.2 How Does This Specifically Relate to Government?

As government budgets shrink, all expenses are scrutinized. As baby boomers retire and as organizations cut costs, positions remain vacant causing the government workforce to decrease. The remaining labor force has taken on more responsibilities and a larger workload. Nonetheless, a gap exists between the amount of work that needs to be completed and the human resources capable of handling that work. In addition to this widening service gap, consumer expectations are disproportionate to funding allocated. Our society expects on demand services 24 hours a day, 7 days a week accessible online. One way governments continue offering public services with less people and to help meet

this societal expectation is by implementing a myriad of new technologies and online services. As this economic shortfall continues, constituents will most likely see online services replace services traditionally offered at local offices, court houses, and satellite locations (i.e. drivers license renewal, vehicle registration or tax filings.).

As new online services are implemented, organizations turn to media and press releases to inform the public. Often social media and networking sites are cost conscience channels of spreading the word to citizens. Government officials are using mediums such as YouTube, Facebook and Twitter to spread awareness of new services as well as events and other announcements. This increase in online services and the advertisement of these convenient resources attracts more hackers to these sites.

While Government Transparency has had positive effects, the threat must be assessed with application security as the core component in the process. While an attack on a government site not holding sensitive information will not result in a direct data breach, it will cause great damage to the constituents using the sites. According to the XSSed project, the largest online archive of XSS vulnerable sites, the Ohio Department of Health site has a cross-site scripting flaw that was identified in June of 2008 which still has not been resolved (XSSed Project, n.d.). The site contains information about health departments, locations, illness advisories, data statistics, and resources. This site's only purpose is to provide information to the public; however it is a known gateway for attackers to exploit Ohio citizens. Even if this issue has been resolved, being listed on the XSSed site will attract undue attention from malicious predators. Constituents should not need to worry about being attacked using this type of resource. These sites exist to serve and protect citizens, not expose users to malicious attacks. It is the government's ethical responsibility to ensure each and every one of their sites is safe, secure, and does not pose a threat.

3. Protecting the Constituents: An Example Utilizing the Kentucky Derby

More than 1.5 million people attend the Kentucky Derby Festival each year (Kentucky Derby, n.d.). Approximately twice the number of people who actually attend a

Derby event actually go online to search for information related to the event such as hotel accommodations, tickets prices, event times, locations, etc. This number can be as high as three million people trolling the internet for public information regarding the Derby. There are many government sponsored websites that advertise and announce Derby Festival information. Chances are most people are not logging into a sensitive web application to gain this information. With this stage set, we must ask the question: What if one of the most popular of these government web applications were riddled with Cross-Site Scripting or a malicious banner advertisement? Here is a step-by-step guide to how a malicious user could compromise the Kentucky Government Derby website.

Step 1: A malicious user has identified a web application vulnerability in a Kentucky Government web application issuing Derby information to the public where user input might exist such as a search box.

Step 2: The malicious user injects a dangerous script into the vulnerable site. At this point, imagination is the only limitation.

Step 3: The user visits the page(s) with the embedded malicious script and becomes the attacker's victim just by visiting an innocuous Derby website

Below explains some of the attack options an attacker has from Step 2:

3.1 Reading/Stealing Cookies

By stealing cookies, the attacker can become another user. For example, the injected script embedded in an image tag sends the cookies to the attacker's malicious site which has referrer logging turned on. The attacker looks for requests with a cookie value in their http logs. This gives them the page that the cookie is valid for. They can then add the cookie to their browser, taking over the user's session. A Kentucky Derby site visitor could browse to a State Parks reservations page to book a cabin during their stay. Since the session is the same and the hacker has stolen the cookie, they could potentially hijack the session and steal the user's credit card information.

3.2 Redirecting User

One very simple and popular use of Cross-Site Scripting is the technique of directing a user to a different site; usually one run by an attacker that has the look and feel of the legitimate site. The victim is tricked into entering some kind of personal information or creating a bogus account so the attacker can then steal information and take advantage of the victim. Using the Derby example, this may be used on a page that advertises ticket sales. Instead of redirecting the user to the appropriate site where a user can buy legitimate tickets, the user is instead redirected to a rogue site in order to steal the users' credit card and billing information. Additionally, the ticket sales site has lost revenue because of the redirect. It is often said that word of mouth is the best way of advertising a business. This is true whether the attention is positive or negative. Celebrities including Kid Rock, Angelica Houston, and even Queen Elizabeth have attended the Kentucky Derby. These people have access to the media and could very easily speak to the masses in regards to their information being stolen due to this type of vulnerability. This could have an extremely negative impact on the Commonwealth of Kentucky's reputation in terms of the lack of security in the website and could affect future funding allocations to the agency responsible for hosting the Derby.

3.3 Harvesting a Client's Browser History

This attack works well with the Derby example as this attack is used for targeted attacks rather than a general mass assault. One of the first things hackers do in a targeted attack is reconnaissance work. Exposure of a user's search history and what applications they may be logged into can give an attacker a lot of profiling information in regards to their victims. This information can be used to break into their corporate networks; personal information such as banking sites and email accounts; or expose private information that might lead to cyber stalking. Many people, despite warnings on the nightly news still use weak passwords. Kevin Johnson used a technique by adding a "note" to his Facebook page asking people a list of questions that are the most common "forgot-password" questions. He masked it by advertising as a "getting to know you better" note. He was able to harvest a large number of password account information. Grossman, Hansen, Petkov, Rager, and Fogie discuss and illustrate several methods for performing this exploit (Grossman, Hansen, Petkov, Rager, and Fogie, 2007) :

- JavaScript/CSS API "getComputedStyle": This technique allows the attacker to use link colors to identify whether or not the victim has visited particular sites. This is done via a targeted query in the address history, but most people use Amazon, eBay, and other well-known sites or email services such as Gmail, Hotmail, etc., so the hacker needs only to write a script querying the most popular sites which incorporate login pages.
- JavaScript Console Error Login Checker: This method can be used to test whether or not a user is logged into a particular site or not. It should be noted that this is Firefox specific, the most popular web browser according to W3Schools.com (W3Schools.com, n.d.).
- In terms of stealing search engine queries, one can build on the JavaScript/CSS API attack above. The results generated from the JavaScript/CSS API attack can give the attacker a view of the victim's interests and therefore, the attacker can now attempt to discover what search queries a user has utilized in a search engine. SPI Dynamics (now owned by HP) came up with a tool called SearchTheft that not only helps illustrate their research in regards to this attack, but also speeds up the process of discovery

itself. SPI Dynamics points out in their research brief, that there are multiple queries constructed based on how a user inputs their search words in the search engine form (SPI Dynamics Labs, n.d.). These queries can differ by case, by order, and by how the user initiates the search (i.e. hitting enter versus clicking on the search button.) SearchTheft is an algorithm that can significantly speed up the iterations of a particular search. They were able to generate "all variations of a six word search query, some 46,080 combinations" in 5 seconds. When an attacker is attempting to perform reconnaissance work in order to find out more about their victim, this algorithm can be a handy way to speed up the process. Many times, if a hacker can find enough information about the person they are attacking; they can soon begin password inference with better defined dictionaries, target them for phishing attacks, and then gain enough information to cyber stalking. For instance, if the victim is a developer, the attacker may find a group they have queried about a problem with a particular application. This may give the attacker a way to not only attack the victim, but enable them to attack another application.

3.4 Taking Over and Controlling User Browsers (Aka Zombies):

Zombies are similar to bots, but zombies run within the browser's context using client side code whereas bots run on the operating system at a lower level. There are various frameworks that can be used as controllers such as AttackAPI and BeEF. These frameworks can connect the user to the malicious server. The attacker can then choose what modules they want to run against that zombie which could retrieve client information, determine what Firefox extensions might be installed, request initiations, inject Javascript to run other attacks, copy clipboard information, and scan ports. In the Derby example, a website visitor may unknowingly expose their entire network mapping thus giving the attacker access to private network folders and files.

- Intranet Hacking: In application security, professionals often refer to pivot points. Pivot points are ways an attacker gets help to escalate an attack to

another level. Web browsing can create pivot points for attackers to get into internal, protected systems. The following image and step-by-step description depicts how this works (Grossman et al., 2007, p. 174):

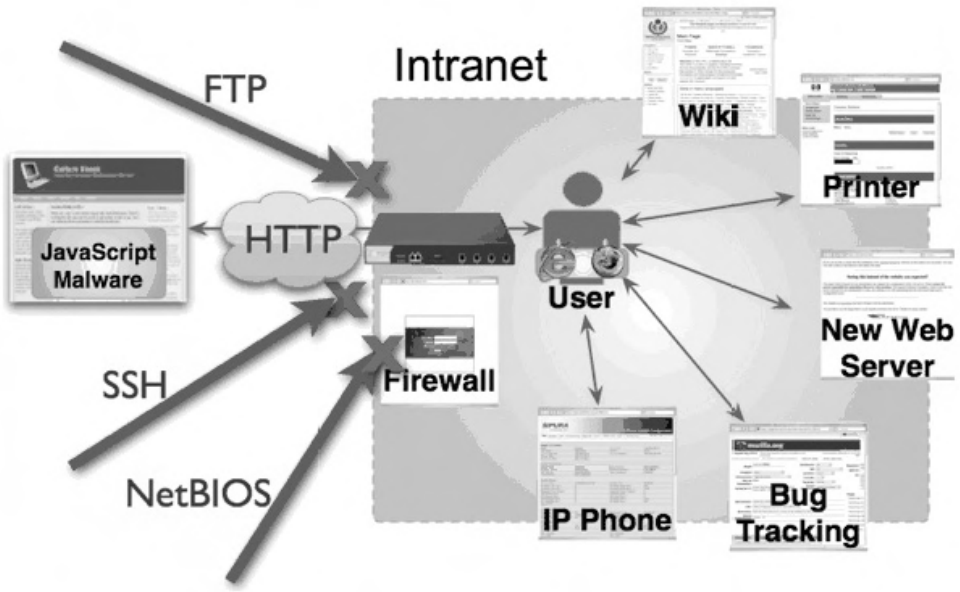

Figure 2: Web Browsing Pivot Attack

Exploit Procedures

1. A victim visits a malicious Web page or clicks a nefarious link; embedded JavaScript malware then assumes control over their Web browser (Grossman et al., 2007, p. 174).
2. JavaScript malware loads a Java applet revealing the victim's internal NAT IP address (Grossman et al., 2007, p. 174).
3. Then, using the victim's Web browser as an attack platform, the JavaScript malware identifies and fingerprints Web servers on the internal network (Grossman et al., 2007, p. 174).
4. Attacks are initiated against internal or external Web sites, and compromised information is sent outside the network for collection. (Grossman et al., 2007, p. 174)

3.5 XSS Defacements

Defacements on web sites have become less of a concern since other types of attacks have matured and can cause far more damage. As with other security defects, old vulnerabilities are still applicable and cannot be totally ignored. In the case of the Derby example, if a rogue user were to exploit a Cross-Site Scripting Vulnerability to deface a government sponsored Derby site, there are consequences that could include negative media, embarrassment, and a general distrust of government sponsored web sites. If there were a highly publicized defacement of a government Derby event site, the constituents might think twice before trusting or using a Department of Revenue website to file their taxes even though the state government sites may be completely different and even maintained by a different entity.

4. Convincing Executives

Prior to a discussion in regards to fixing the code itself one must discover a way to convince executives, managers and developers of the importance of coding against web attacks on public applications. Though regulatory compliance helps to back up the security expert's justification, by itself, it is not enough and one has to keep in mind that being compliant is not the same as being secure. Looking back at the history of cross-site scripting helps identify how people made the wrong assumptions about the consequences of such an attack. Microsoft and CERT (Computer Emergency Response Team) published a paper in regards to cross-site scripting on February 2, 2000. Despite the publicity of the problem and the fact that malicious hackers were using it to wreak havoc on a small scale, developers tended to ignore the issue. All of this changed in October of 2005 when the Samy Worm hit MySpace. The virus had a widespread influence as it affected more than a million users. Hackers especially saw the implications and began to use XSS in much more devastating ways. The assumption that firewalls and SSL protect web servers has been proven to be a fallacy (Grossman et al., 2007). In order to get executives to realize the importance, security professionals must build a case within their organization. Putting together a concise portfolio of case studies, whitepapers, power

point presentations, and newsletter articles from outside security consultants and analysts can illustrate the point and help executives become aware of the breaches that have occurred at partner institutions or sister agencies. For instance, the state of Ohio has made many headlines due to the numerous breaches that have occurred over the past few years. While the state of Ohio has worked very hard to dramatically improve their situation, they serve as a good example to other states. The point is to convince the decision makers that they need to be proactive and learn from the mistakes of others. Turn the argument of "it will never happen to me" into "I don't want to be like them." In addition, developing business value metrics and prioritizing sites will help you meaningfully quantify the business value proposition of risk mitigation and help show executives where their precious dollars should be spent.

By developing a concise portfolio of application security awareness documents, compiling case studies related to comparable organization breaches, conferring with security experts on the state of risk within your organization, and implementing a business value metrics initiative of all sites you can open up a dialog with executives and decision makers about implementing preventative measures.

5. Prevention

It's important to point out that since public web servers advertise themselves to the world via port 80, there is little protection from malicious intent except for what the developers implement in the code. Web application firewalls may offer a short-term solution that minimize some of the risk; however the key to application security lies in writing secure code. One of the problems associated with this change in web application security is that developers are held to strict deadlines to create a site that functions. Often times proper security testing isn't implemented. Developers tend not to think about how someone could break their code. There is an old story about a computer science student who built a calculator application for a class. The student wrote the code and tested it to the point they were sure they would receive an A. When it was time to present the new application to the professor, the first thing the professor did was enter A+B and the

application simply did not know how to process the input, so the application failed. Every developer can learn a lesson from this simple example.

There are simple preventative measures we can employ in order to help avoid problems like the cross-site scripting examples above in the previous section. The most important thing we can do is to educate our developers and implement secure coding standards and code reviews. The SDLC (Software Development Lifecycle) as well as ITIL (Information Technology Infrastructure Library) instruct organizations to implement security from the beginning of a project kickoff through the implementation. This not only includes new projects, but enhancements to applications. It is imperative to have input from the application security professionals in an organization from the onset of a project. This is because it is very difficult to inject secure code into an application after it has been written and/or designed. It is also much more expensive to retrofit security directly before implementation. If an application is written without regards to security, the developers may have to change hundreds of lines of code. With tight deadlines in place, problems may get implemented without being fixed at all.

If developers are praised after a public facing site is published sans major security issues, the management team and security personnel can point to that example as a good practice for future maintenance and development projects This practice can also help to strengthen the relationship with between the developer group and security group. For instance, if a particular Derby website hosted by the Governor of the Commonwealth of Kentucky is usually visited by 1.5 million people, it should be noted that the developers responsible for implementing such a safe site have helped to protect 1.5 million people.

No development effort should be taken lightly when looking at simple business rules. It is up to ethical development standards to build every site so that it can be maintained with regard to security. Enterprise committees can help by creating minimum level secure coding standards and review processes for all web applications. These committees can even develop a library of sub-routines which each agency can utilize in their own applications. These libraries of ready-made code will help with the consistency of the security program and ease efforts of remediation by saving time for both the developers and the security specialists. If enterprise committees don't exist at your organization, work with executives across departments and cabinets to organize a strong

team of decision makers to expose the problems of web application attacks and the need to allocate funding to address these concerns.

The key is to "bake security in instead of brushing it on". It should never be assumed that a public facing website will always hold only public data. Sites grow and evolve constantly. As business rules change, new political appointees are elected, and development groups grow or wane, systems can become repurposed or combined. This means that what yesterday was a simple application to advertise Derby Festival events may morph into something completely unexpected by the original developer. With this in mind, here are a few general prevention methods to keep applications safe.

First, reliable server side input validation should be incorporated into code in a layered approach (i.e. whitelists followed by blacklists and HTML encoding). This method can prevent many currently known vulnerabilities as well as any new attack vectors that have not yet been discovered. Additionally, encryption should always be included wherever userids and passwords are utilized. It is also important to keep a detailed list (version, implementation date, etc.) of any and all open source pieces or parts (i.e. plugins, APIs, etc.) and make sure they stay patched whenever a vulnerability is published. Lastly and most importantly, never assume vendor solutions are secure. Always asses newly purchased software in house and/or by a third party. If vulnerabilities are identified, report them to the vendor service desk to seek remediation. Once the vendor publishes a patch or update to resolve the issue, follow up with another assessment. It is also advisable to always test software in your own environment or gain the appropriate permissions and legal agreements if the application exists in an environment outside of your control.

6. Conclusion

In order to fight against cyber enemies who are becoming more and more efficient, the security industry has to work harder to keep up with attackers. While we have improved at protecting sites transmitting sensitive data by enhancing our infrastructure, creating robust firewalls, and implementing solid input validation, it is time to focus on the weaknesses of our public facing sites which transmit no sensitive data and could serve

as gateways to sensitive information and networks. It is time to show customers and colleagues how attacks like cross-site scripting can be far more harmful than simply stealing sensitive data. Public facing sites pose as much, if not more, of a threat due to their inconspicuous nature. The government especially has a legal and moral obligation to protect the constituents from being unsuspecting victims of malicious attacks as more and more services are offered online, despite or because of budget cuts.

There needs to be a cultural change and this type of change is the hardest to implement in hard economic times. With tight budgets, making a case to executives to spend funding on preventative measures to secure public facing sites is very difficult. Security professionals need to build a case within their organization for the prioritization of application security funding. It is imperative that agencies become proactive instead of reactive when it comes to application security. Learning the hard lessons after a breach has taken place is too late. The public has watched businesses go bankrupt over the cost of an individual breach. Government is not immune to this kind of economic hemorrhaging as we saw with the Virginia Prescription Monitoring Program and the state of Ohio's security breaches. As custodians of the citizens' data and providers of public service, there is an ethical responsibility that can no longer be ignored. In the long term, instilling secure coding practices, processes, resources, and standards will save the government money. The culture change is based on the same principles the insurance business relies on. Though an individual may never suffer from a house fire, the majority of people still invest in fire insurance. Web application security concepts are of the same nature. Anyone who implements an insecure application is merely playing the odds just as a Derby attendant bets on horses at Churchill Downs.

Kristen Sullivan, Kristen.sullivan@ky.gov

7. References

Claburn, Thomas. (August 2008). Most security breaches go unreported. Information Week. Retrieved December 22, 2010, from World Wide Web: http://www.informationweek.com/news/security/attacks/showArticle.jhtml?articleID=209901208

Condon, Stephanie. (September 2010). Tea party supported by one in five in new CBS news/NYT poll. CBS NEWS. Retrieved December 22, 2010, from World Wide Web: http://www.cbsnews.com/8301-503544_162-20016526-503544.html

Government Computer News. (July 2010). Great gov web apps: 10 examples of innovative, practical web-based tools that deliver effective services, collaborative environments and data transparency. Retrieved December 8, 2010, from World Wide Web: http://gcn.com/Issues/2010/07/July-19-2010.aspx

Grossman, J., Hansen, R., Petkov, P.D., Rager, A., & Fogie, S. (2007). Cross site scripting attacks: XSS exploits and defense. Burlington, MA: Syngress Publishing, Inc.

Harlow, David. (May 2009). The Virginia prescription record security breach: the big picture, and using this case as a learning experience. Health Blawg. Retrieved December 22, 2010, from World Wide Web: http://healthblawg.typepad.com/healthblawg/2009/05/the-virginia-prescription-record-security-breach-the-big-picture.html

Jackson Higgins, Kelly. (June 2010). Gartner: Security spending to drop this year and next. Darkreading. Retrieved December 22, 2010, from World Wide Web: http://www.darkreading.com/security/security-management/225600254/index.html

Kentucky Derby. (n.d.). Kentucky derby festival: so much to celebrate!. Retrieved June 15, 2010, from World Wide Web: http://www.kentuckyderby.info/festival.php

Pew Research Center for the People & the Press. (April 2010). Distrust, discontent, anger and partisan rancor: The people and their government. Retrieved December 22, 2010, from World Wide Web: http://people-press.org/report/606/trust-in-government

The Privacy Rights Clearinghouse. (n.d.). Chronology of data breaches. Retrieved December 22, 2010, from World Wide Web: http://www.privacyrights.org/data-breach/new

Raywood, Dan. (January 2010). Adding a second protective layer and effective correlation is the best defence against cyber attacks. SC Magazine: Secure Business Intelligence. Retrieved December 22, 2010, from World Wide Web: http://www.scmagazineuk.com/adding-a-second-protective-layer-and-effective-correlation-is-the-best-defence-against-cyber-attacks/article/161616/

SPI Dynamics Labs. (n.d.). Stealing search engine queries from JavaScript. Retrieved June 15, 2010, from World Wide Web: http://packetstorm.austin2600.net/papers/web/JS_SearchQueryTheft.pdf

U.S. Department of Health & Human Services. (n.d.). Health information privacy. Retrieved June 15, 2010, from World Wide Web: http://www.hhs.gov/ocr/privacy/hipaa/administrative/breachnotificationrule/postedbreaches.html

W3Schools.com. (n.d.). Web statistics and trends. Retrieved December 22, 2010, from World Wide Web: http://www.w3schools.com/browsers/browsers_stats.asp

Winkler, Ira. (2005). Spies among us. Indianapolis, IN: Wiley Publishing Inc.

Kristen Sullivan, Kristen.sullivan@ky.gov

XSSed Project. (n.d.). XSS achive. Retrieved December 24, 2010, from World Wide Web: http://www.xssed.com/mirror/42812

Kristen Sullivan, Kristen.sullivan@ky.gov

Last Updated: June 24th, 2017

Upcoming SANS App Sec Training

SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
Community SANS Minneapolis DEV534	Minneapolis, MN	Aug 25, 2017 - Aug 28, 2017	Community SANS
Community SANS San Francisco DEV541	San Francisco, CA	Aug 28, 2017 - Aug 31, 2017	Community SANS
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Secure DevOps Summit & Training	Denver, CO	Oct 10, 2017 - Oct 17, 2017	Live Event
SANS Seattle 2017	Seattle, WA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Cyber Defense Initiative 2017	Washington, DC	Dec 12, 2017 - Dec 19, 2017	Live Event
SANS Cyber Defense Initiative 2017 - DEV522: Defending Web Applications Security Essentials	Washington, DC	Dec 14, 2017 - Dec 19, 2017	vLive
SANS Security East 2018	New Orleans, LA	Jan 08, 2018 - Jan 13, 2018	Live Event
SANS OnDemand	Online	Anytime	Self Paced
SANS SelfStudy	Books & MP3s Only	Anytime	Self Paced