

SECURICON

Information Security Solutions

Deploying SSL to protect your web applications?

What you might be missing...

Authors: Frank Dimina & Brad Geesaman

frank.dimina@securicon.com

brad.geesaman@securicon.com

www.securicon.com

Background

This whitepaper is based on the observations of professional information security consultants who have performed numerous vulnerability assessments for both commercial and government organizations. It is reasonable to assume that a wide range of corporate and government networks are exposed to issues described here. This information is presented in the interest of raising awareness among organizations to help them properly defend against a new trend of threats and attack vectors.

Summary

Protecting your enterprise's information security assets is akin to a military arms race. Security managers must constantly research and deploy new tools and technologies while maintaining a constant awareness of new threats and attack vectors. However, as soon as security practitioners achieve adequate levels of protection for a given attack surface, hackers adapt and improvise their tactics to find new avenues into the victim's network. It is truly an arms race—one where security managers must stay at least one step ahead of their adversaries or risk finding themselves caught off-guard when the next big threat occurs. Unfortunately, the security teams defending these networks will always be at a disadvantage. Success can only be achieved by eliminating every possible security hole, while attackers only need to find a single vulnerability to accomplish their goals. It's readily apparent that the odds are in favor of those with malicious intentions.

With the maturation of processes and controls surrounding network security, the success of massive network-based attacks has declined significantly. Hackers have found that targeted attacks against web applications are now more lucrative. In addition, many companies who deploy common security tools, such as Secure Sockets Layer (SSL), with the intentions of securing their web applications may actually be increasing their risk by removing their ability to detect malicious content hiding in their web transactions.

The Changing Threat Landscape

Until recently, adequate security could be achieved by deploying a “defense-in-depth” security model that provided multiple layers of cascading security controls throughout the network. However, network vector attacks are becoming increasingly ineffective. We can attribute a significant portion of this decrease in successful network attacks to the improvements and maturation of most organization's security practices, tools, and techniques. However, attackers no longer need to spend hours trying to find a single hole in your firewall policy. They can walk right into your network through the front door, via your web applications. If this intrusion occurs over a SSL connection, you probably will not even see the attack occurring.

Over the past 12-24 months, Securicon has observed a shift in the primary attack vectors used to penetrate an organization's information systems. Our work has shown that attackers have changed their focus and are now concentrating the bulk of their efforts on the new low-hanging fruit -- your web applications. The scary truth is that most applications are not designed to resist intrusion, and these attacks are becoming extremely lucrative.

The state of web application security today is comparable to that of network security ten years ago. Many system owners still do not fully understand the risks facing their web applications and may not be

SECURICON

aware that their applications, users, and data are exposed to serious vulnerabilities. Additionally, products and technologies designed to protect the application level are typically immature, complicated, and costly. It's no surprise that they are often misused and misunderstood.

To make matters worse, many security teams have become complacent in their efforts to defend their systems against application level attacks. The maturation of network level defenses has led to a false sense of "total" security for most organizations. While the past couple of years may have seemed quiet to some incident response teams, let us not assume that the hackers have ceased their efforts. Their lack of noise is more likely due to the fact that they are focusing their efforts on new targets that allow them easier access into your organization and data while bypassing your capabilities to detect and defend against these attacks.

Attackers are leveraging this complacency. They are doing it right now, on your network, with your web applications, targeting your users. And they are doing it using security controls that you thought you could always count on, such as Secure Sockets Layer (SSL). The remainder of this paper will illustrate how organizations that place too much confidence in SSL could be leaving their web applications open to some very significant security risks.

Deploying SSL securely, what you might be missing?

Nearly every single security consultant has heard the following statement from a client: "We have SSL on our web applications, so they are secure." Unfortunately, this is not an exaggeration. SSL has been widely misunderstood and misused for years. If used properly by both users and system owners, SSL can provide some true security benefits, but these benefits come with their own unique drawbacks. When used improperly or in an insecure setting, SSL can actually *decrease* the overall level of security on your systems by limiting the ability to detect and prevent application level attacks.

Confusion over SSL as an effective security tool is reminiscent of the early days of the VPN industry. For years, organization's would state, "We have a VPN, so we are secure" and it took many more years (and many painful lessons learned) for organizations to realize that VPNs are a secure remote access tool, not a security tool. In fact, many security experts maintained for years that, if not implemented appropriately, VPNs created additional security *risks*, as they extended an organization's network borders and were often implemented with weak authentication mechanisms. It was not until the attackers began fully exploiting these weak VPN implementations that organizations finally understood the inherent risks and began operating them securely by adding the appropriate controls to gain their full benefit without impacting overall security.

We are currently repeating this costly lesson now with SSL. The widespread use (or misuse) of SSL without additional compensating controls, combined with ubiquitous vulnerabilities in web applications have created significant security exposures to organizations and users across the globe. The following sections of this whitepaper will provide examples of how attackers are using SSL to compromise your data, along with some recommendations on how you can help protect your applications and users from these attacks.

To be clear on one point: SSL itself is not inherently risky, but it's the way in which SSL is used, or can be abused by attackers to shield attacks from observation, that creates this high level of risk.

A False Sense of Security

Unfortunately, the majority of web applications are still vulnerable to Cross-Site scripting attacks¹ (XSS). It's quite rare for Securicon consultants to find web applications that do not have multiple XSS vulnerabilities present throughout the system and its various web interfaces. While some organizations downplay the risks of XSS attack vectors, attackers have seized upon the widespread availability of these vulnerabilities. The risks of XSS extend to an application's users as well. XSS attacks can be used in phishing and spam campaigns to send an organization's customers hyperlinks containing advanced attacks. These attacks can be used to steal user's login information, capture account information, proxy communications through an attacker's system, or even run custom code on a customer's system without their knowledge.

How does SSL fit in? SSL makes it even easier for an attacker to use XSS attacks to steal a user's login information while maintaining the appearance of a valid, secure site. Let's assume that a hypothetical banking site is vulnerable to XSS attacks. The bank's application developers have implemented a valid SSL certificate on the login page to help give their customers assurance that they are indeed connecting to the legitimate banking application. When logging into the web site, the user's browser presents them with the "Lock" symbol, letting them know that the application has a valid certificate and that the communications are encrypted (and that the server is verified to be owned by the bank). Along comes the attacker. Let's assume that the attacker has already identified that this banking site's login page is vulnerable to XSS attacks. Using this vulnerability, the attacker creates a malicious URL with embedded scripts that replaces the login box (usually in its own frame on the login page) on the rendered application login screen with his own login box that looks identical to the main site, but it passes all usernames, passwords, and PINs to his own system. The attacker then distributes this malicious URL to the bank's customers. When the victims use the malicious URL to visit the bank's login page, the embedded scripts capture their credentials and silently send them off to the attacker's system. With valid credentials and sessions, the attacker can now login and liquidate their account balances.

In most cases, the banking customers would be unaware of this attack until it is too late. The users were simply obeying the "good surfing" habits that administrators have been preaching for years, such as only using sites that start in "https://" and that have the browser "lock" symbol. However, the problem with these attacks is that the SSL "lock" symbol is now giving users a false sense of security. To the user, the compromised application looks legitimate, and the server conveys the message that it can be trusted due to the valid SSL certificate. Even if the browser warns the user that "some items on the page are not encrypted", many users will actually ignore that warning and continue transacting with the site. This attack scenario is actually quite simple to execute and does not require a technical mastermind to conduct. With the abundance of web sites vulnerable to XSS attacks, it is highly likely that an attacker will find some financial or shopping applications that could be exploited to expose your sensitive financial information such as bank accounts or credit card numbers. It's only a matter of time before the users that are compromised by these attacks decide to hold the financial application providers liable, especially since the site provided a valid, secure SSL session.

¹ Cross site scripting (XSS) is an exploit method where malicious web code or web content is inserted into a trusted or alternate web site or web application to run the attack or exploit code in the context of the trusted or alternate site.

What You Don't Know Will Hurt You

In many scenarios, SSL communications actually shield an attacker's actions, enabling them to conduct unrestricted attacks with little fear of detection. Instead of providing an extra layer of defense, SSL actually provides a benefit to the attacker. Using SSL channels to hide attacks from detection has become so popular that most attackers are ecstatic to find SSL available on their targeted systems and applications.

Most public facing web servers are placed in segmented networks protected by firewalls and Intrusion Detection/Prevention Systems (IDS/IPS) as a part of the standard "defense-in-depth" policy. Typically, the only services open to the Internet are SSL-enabled connections. Because the encrypted SSL connection from the client browser terminates directly on the web server, the entire contents of the connection are not visible by the network IDS/IPS. Attackers are provided with an encrypted tunnel through which they can carry out their attacks against the server, while the entire communication is encrypted by the SSL tunnel. Without knowing it, many organizations have lost the ability to detect and respond to malicious attacks against their web applications. These organizations may have rendered their IDS systems completely "blind" to the attacks occurring against their systems.

If the bank in the previous scenario implemented their site in this manner, the attacks may have gone unnoticed until the victims notified the bank regarding the missing funds. Even worse, the web server logs may only have bits and pieces of the attack on record due to the nature of these attacks. Hackers are keenly aware of this situation, and they have been continuously finding new ways to exploit web browsers' built-in trust model and creatively combine them with common web application vulnerabilities to carry out highly successful and damaging attacks.

What Can We Do?

This paper is not intended to serve as a groundbreaking announcement of new, cutting-edge attack methods. The overall purpose of these observations is to educate security management teams on attack vectors that are in use today and are quickly becoming the norm. To help deal with these risks, there are several recommendations offered below that organizations should consider to better protect themselves against these types of attacks.

1. Perform routine web application security assessments to eliminate vulnerabilities such as cross-site scripting, SQL injection, and other threats targeted at the application level. Organizations need to seriously address cross-site scripting vulnerabilities in their applications and ensure that developers are trained to avoid adding new ones as updated software is rolled out into production.
2. Increase awareness of the proper methods for implementing SSL and educate users on the newest types of attacks. Help train your customers and users to not ignore SSL warning errors from their browsers. Educate your customers on the seriousness of these browser warnings and ensure they do not click through the messages.
3. Restore IDS/IPS visibility by offloading SSL functionality from the web server to a dedicated SSL concentrator appliance on the edge of your network perimeter. Client and customer communications traffic should traverse the application provider's directly connected network in clear-text so that the IDS/IPS system can monitor the connections and data for attacks.

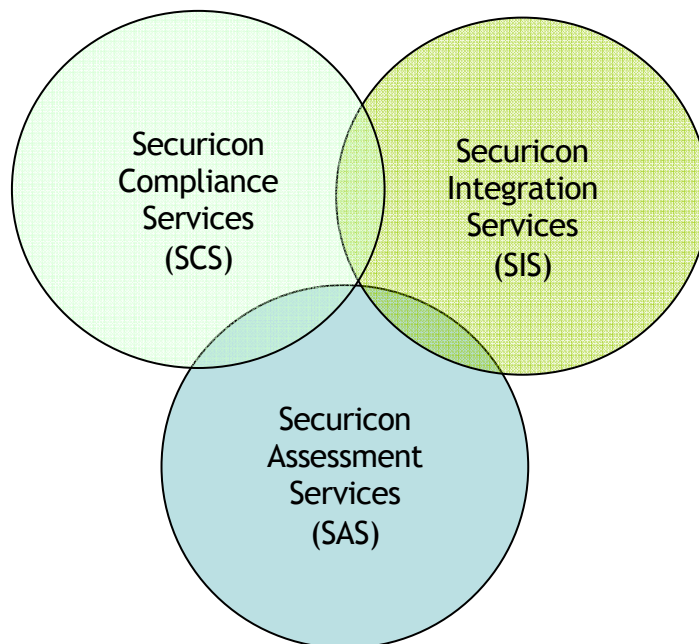
SECURICON

4. Consider using an effective Host-Based Intrusion Detection/Prevention System (HIDS/HIPS) on critical web applications and servers. The HIDS/HIPS must be capable detecting exploits occurring through SSL connections against the server and/or web application.
5. If you are using a managed security services provider (MSSP), ask your vendor to ensure that their monitoring systems are positioned to see all traffic to/from your applications. Perform an audit of the MSSP's monitoring capabilities to ensure all attacks are visible to the service provider's analysts, especially those occurring over SSL channels. IDS systems that are solely monitoring encrypted SSL traffic might as well be turned off.
6. Finally, the security industry needs to address the problems with SSL and provide new solutions that are customer-enabling. The current tools and technology are difficult for the average Internet user to understand. The average application user should have tools that give them a better sense of a web server or web application's identity and level of security. The browser "lock" icon is now an inadequate protection mechanism that is dire need of an upgrade. Microsoft has taken some positive steps in this area with new browser warnings in Internet Explorer 7 that block the site and provide the user with a clear, strong warning before when SSL inconsistencies are detected. Until better tools are available, the existing warnings will need to be supplemented with increased consumer and user security awareness and education.

About Securicon

Securicon's unique approach to professional services allows for the flexible delivery of cost effective security solutions to complex client problems. Securicon consultants are network, system and application security experts that provide the following core services to our customers:

- Security Compliance Services (SCS)
- Security Assessment Services (SAS)
- Security Integration Services (SIS)



In addition to application, network and system security experience, Securicon engineers have in-depth expertise in critical infrastructure environments, including extensive experience in the financial services and power and energy industries. For more information, please visit www.securicon.com