AppliCure Technologies

dotDefender

Web Application Security

Web Application Security 101

Web Application Security 101

As the Internet has evolved over the years, it has become an integral part of virtually every aspect in the business process cycle. In the early days of the Web a company's online presence consisted of a static Website that promoted products and provided visitors with company information. The emergence of certain technologies like AJAX, PHP, and Document Object Models gave businesses the ability to move from placing nothing short of a company brochure on the Web to deploy dynamic, feature-rich applications that drive sales through e-commerce; provide online services to their employees; establish open ended communication between themselves and their customers; and allow for collaboration among employees, partners, suppliers, and clients.

In addition to providing employees and customers with a more dynamic experience, Web applications have become a way for businesses to save money. By turning to Software as a Service and cloud based solutions, organizations have found that they are able to trim their budgets by:

- Spending less on resources such as servers and networking infrastructure
- Reducing power consumption and related costs
- Avoiding capital expenditures associated with IT
- Using technology that is flexible and scalable

Of course, as the usefulness and complexity of the Internet grew through increased use of Web applications, the security risks involved also grew proportionately. To combat the threats that these applications face, many organizations look towards traditional network security solutions. Thinking that deploying a network firewall, intrusion detection system, or intrusion prevention system works to protect the network perimeter from attack at the application layer (OSI Layer-7) can be a huge mistake.

The traditional approach to network security aims to protect resources such as servers, workstations, printers, internal databases, and other network resources. The tools used to secure these resources work by preventing access to certain ports or services by creating allow or deny rules to network packets. Blocking access to port scans, worms, viruses, and other attacks aimed at networking protocols works to prevent intrusion over OSI Layer-3, but does nothing to prevent the sophisticated attacks that take place at the application layer because their simple approach does not work in an environment where each application differs from another.

> ***Many IPS systems cannot even look into simple SSL encryption, and are, therefore, relegated to blindly forwarding SSL traffic without inspection.***

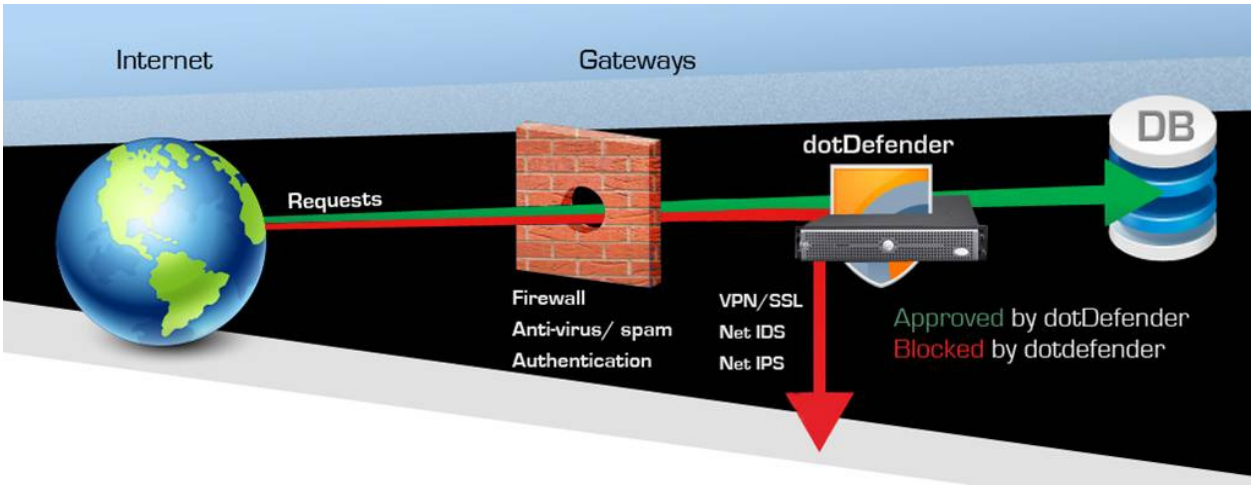

Figure 1. Traditional firewalls keep out malicious network traffic but malicious Web traffic pass through freely.

Web application security relies on the ability to inspect HTTP packets to handle threats at Layer-7 of the OSI model. Attackers are all too familiar with the fact that traditional perimeter security methods do not stop attacks against Web applications that are, by nature, designed to allow visitors to access data that drives the Website. By exploiting simple vulnerabilities in Web applications, an attacker can pass through perimeter security undetected accessing data and even the network your traditional firewall and IDS systems are in place to protect.

> ***According to a Gartner study, 75% off all attacks on Web sites target the application level and not the infrastructure.***

Understanding the Risks

To help IT professionals better understand the security risks that surround Web applications, a community of concerned individuals created the Open Web Application Security Project, or OWASP for short. In addition to a collection of open source tools, training and projects, OWASP publishes a list of the Top Ten Risks to Web Application Security. Among the most prevalent threats to Web applications are:

- Injection attacks (1)
- Cross-site scripting (2)
- Security misconfiguration (6)
- Failure to restrict URL access (7)

Injection attacks are the result of a Web application sending untrusted data to the server. The most common attack occurs from malicious code being inserted into a string that is passed along to a SQL Server for execution. This attack, known as SQL Injection, allows the attacker access to data which can be stolen or manipulated. Other types of injection attacks include Code Injection and Carriage Return/Line Fee (CRLF) Injection.

Cross-Site Scripting, or XSS, is the most prevalent security flaw that Web applications are vulnerable to. In an XSS attack, the attacker is able to insert malicious code into a Website. When this code is executed in a visitor's browser it can manipulate the browser to do whatever it wants. Typical attacks include installing malware, hijacking the user's session, or redirecting a user to another site.

Security Misconfiguration is the result of poor administration of the Web server or application server and often leads to path traversal vulnerabilities. Allowing unauthorized or unprotected access to files, directories, or accounts can lead to an attacker completely compromising a system that is vulnerable.

Failure to protect URL access is another flaw that allows attackers to exploit the path traversal vulnerability. Only in this case, the attacker simply amends the URL to see if he is granted access to a private page or directory within the Website.

Statistical Data

Just how many Websites are vulnerable to the different attacks? According to the Web Application Security Consortium's most recent Web Application Security Statistics Project, the probability to detect a vulnerability classified as either urgent or critical in a Web application is 96% if done by white box testing. So practically every Website that runs an application is vulnerable to some type of attack. Finding it is simply a matter of determination on the part of the attacker.

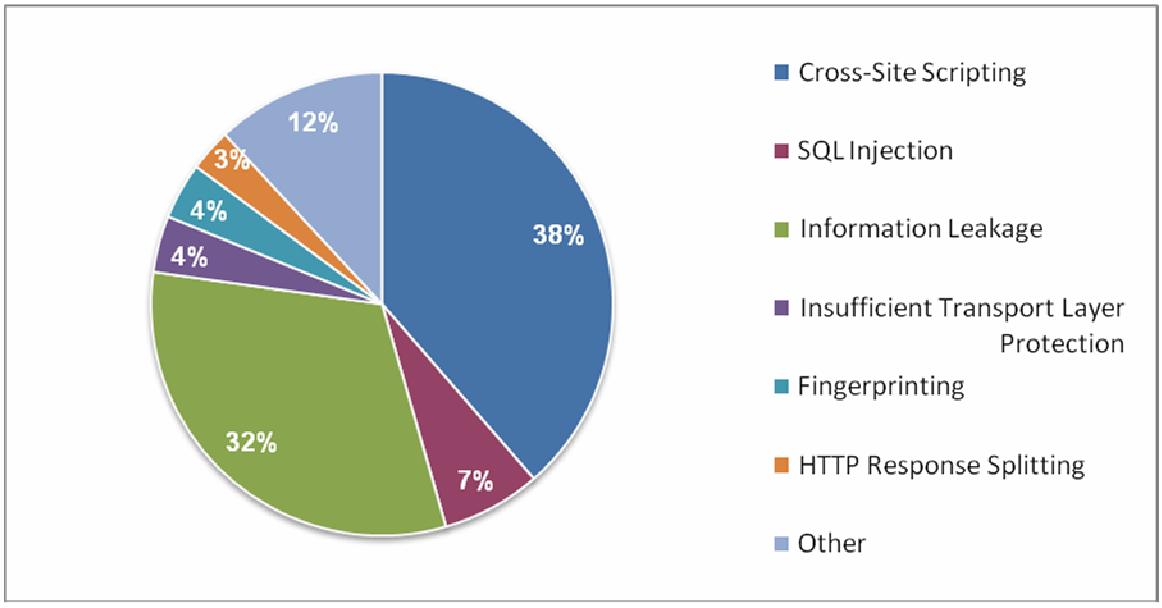

Figure 1 - The most widespread vulnerabilities found in Web Applications (source: WASC Web Application Security Statistics Project).

According to these numbers, 39% of all vulnerable Web applications are susceptible to Cross-Site Scripting attacks. SQL Injection, on the other hand, is found in only 7% of all vulnerable Web applications. So why are injection attacks at the top of the OWASP Top Ten list? For two reasons:

1) Several types of injection type attacks are included in the category.

2) The Top Ten list reflects the risks associated with each vulnerability, not only the saturation. Injection attacks are much more dangerous because, especially with SQL Injections, they allow the attacker direct access to data. Whether it be authentication data, health records, card holder data, or any other confidential information the ability for the attacker to access this makes injection attacks the most dangerous risk to Web applications.

The most widespread vulnerabilities found in this report can be easily aligned to the OWASP Top Ten risks we just discussed.

WASC	OWASP
Cross-Site Scripting	**Cross-site scripting**
SQL Injection	**Injection attacks**
Insufficient Transport Layer Protection	**Failure to restrict URL access, Injection attacks**
Fingerprinting	**Security Misconfiguration**
Information Leakage	**Security Misconfiguration**
HTTP Response Splitting	**Injection attacks, Cross-Site Scripting**

Table 1: Aligning WASC to OWASP.

More interesting may be the percentage of Web applications that are vulnerable to the different types of attacks. In the following chart, the data shows the probability, by percentage, that the most widespread vulnerabilities will be found on a Website.

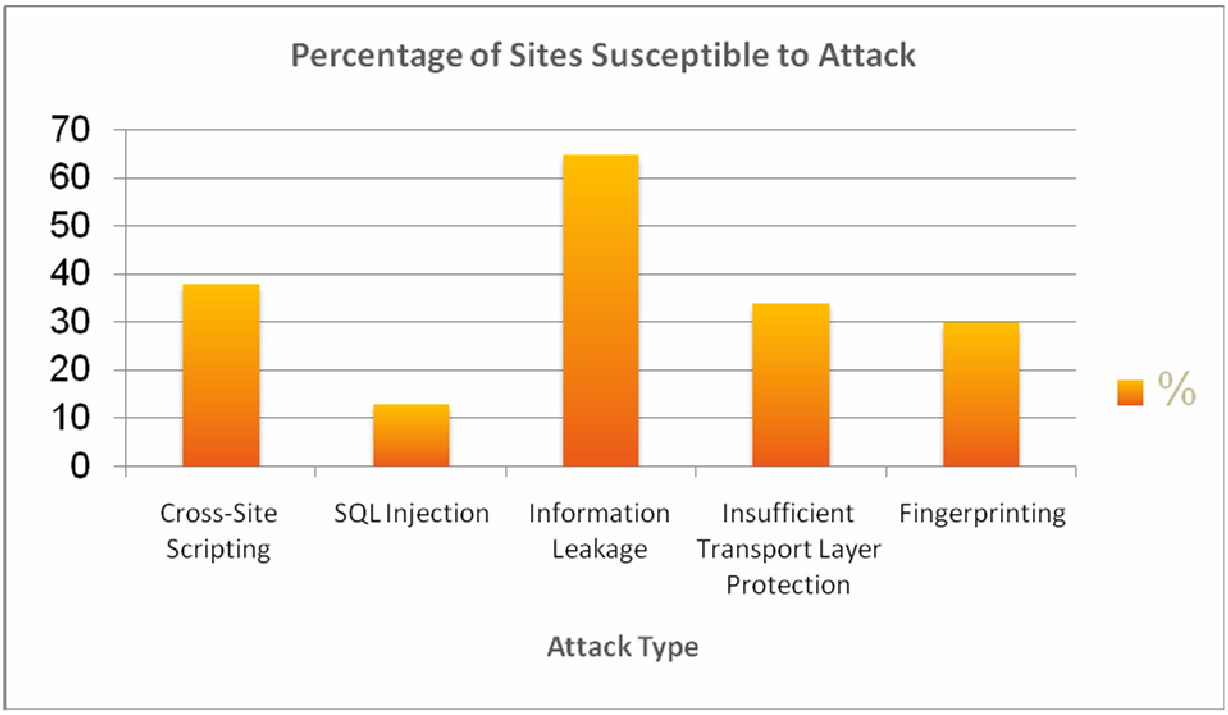

Figure 2 - The most widespread vulnerabilities and the probability that a Website contains it.

Clearly, the numbers total much more than 100%, showing that many Websites are vulnerable to multiple exploits.

Automated Attacks

Based on the sheer number of vulnerable sites, attackers have taken a different approach towards attacking them. Unlike the tedious hours spent hacking a network's perimeter, attacks against Web applications can be easily automated - and with the help of a bot net, large scale, coordinated attacks against multiple sites can net the cyber criminal millions of dollars with a minimal amount of work or skill.

By visiting any number of hacker forums, the attacker can locate specific code strings found in vulnerable Websites. Performing an automated search for this string can collect lists of Websites that contain a specific vulnerability (this is the fingerprinting process mentioned earlier).

Once the attacker has accumulated a list of potential victims, they can download a tool to use in launching the attack. Again, a bot net can be used to increase the pool of targets since the attack is automated. The attacker simply collects the information he or she is looking for and moves on to their next attack.

The Aftermath of a Successful Attack

According to the Computer Security Institute's Annual Computer Crime and Security Survey, the average cost per incident is $300,000. Don't be fooled into thinking that this number represents only the largest organizations. Close to a quarter of all those surveyed in this report were organizations that have

between 1 - 99 employees. To a large company, losing $300,000 dollars can put a dent in the bottom line, but to a small business this loss can be devastating.

Web applications are used to process data and make it accessible to users across the Internet. Whether the application is used to process credit cards, manage employees, or increase collaboration between partners, failing to protect these applications can have serious ramifications.

Compliance Issues - To ensure that organizations do what is necessary to protect confidential information, governments and industries alike have put in place certain requirements. US laws like the Healthcare Information Portability and Accountability Act (HIPAA) are used to protect confidential health information. To protect credit card information, the Payment Card Industry (PCI) has created its own set of requirements. Failure to comply with industry wide or governmental requirements often result in large fines waged against the organization responsible for protecting the information.

Data Theft - In addition to personal and financial information being stolen, a large cost that organizations may face after having a Web application compromised is the loss of proprietary information. Web applications process a great deal of personal information; however, they are also used for collaboration and project management. Organizations with offices across the globe need some way of working together and many Web applications provide such a way. Attackers know this, and often times the objective of their crime is to steal intellectual property either as corporate espionage or to sell to a competitor.

Customer/Visitor Loss of Trust - This is one of the most intangible costs associated with a Website being attacked; it is, at the same time, one that should be expected. When the TJX Companies made the news because they were victimized by Albert Gonzalez, customers lost trust in their ability to protect their credit card information. Likewise, when Google warns that a Website is untrustworthy, even the most loyal visitors avoid the site for fear of having their information stolen or their computer infected.

Burden on Resources - Not all attacks are launched with the intent of profiting directly. Attackers still launch Denial of Service attacks against Websites to disrupt service to legitimate visitors. Additionally, compromised Web servers and sites are used to host multimedia files, malicious files, and links to other Websites without the knowledge of the owner. In these instances, storage space and bandwidth are wasted on illegitimate use.

Ability to Attack the Internal Network - Those organizations who host their Web servers on site risk having one of their applications serve as a entry point to the internal network where other servers, databases, and computers can also be compromised. Going right through the vulnerabilities in these applications bypasses any network perimeter defenses put in place.

The dotDefender Solution

Earlier, we saw how traditional network security solutions do not effectively protect against the common vulnerabilities that exist within a Web application framework. However, because these tools do not adequately protect against Web application vulnerabilities doesn't mean that there is no defense

against these threats. On the contrary, a Web Application Firewall solution like dotDefender provides protection that meets compliance regulations set by one of the most stringent industry security standards there is, the Payment Card Industry Data Security Standard.

A Web Application Firewall or WAF, is an appropriate solution to defend against the common avenues of attack used against Web applications. Deploying a WAF is like placing an eavesdropping agent right next to the Web server itself – serving as a two-way filter that prevents malicious requests from reaching the Web server while at the same time sifting through the responses provided by the Web serve to weed out sensitive or personally identifiable information. The WAF thus serves to not only defend against attacks, but also to mitigate the potential for information leakage. The immediate proximity to the server means it is the last stop in information flow – right before the request must be served, but well after precursor steps such as encryption and fragmentation.

Intrusion detection/prevention systems are effective solutions for protecting the network perimeter, as are traditional firewalls; however there are some distinct characteristics that make Web Application Firewalls effective in protecting what other solutions can't - the application layer.

In a network firewall, complications at the packet level like encryption and fragmentation pose a significant, often insurmountable challenge. However, a WAF remains blissfully oblivious to these complications, and is free to focus on what it understands so well – application level security.

The way WAFs handle application layer logic differentiates them from intrusion prevention and detection systems as well. In addition to having an understanding of protocols, WAFs also recognizes language patterns, such as XML, SQL, JavaScript, HTML, PHP, and many others.

How dotDefender Works

dotDefender is a software based Web Application Firewall that when integrated within an existing Web server begins to protect your Web applications almost immediately. Using three different security engines, dotDefender is prepared to take proactive steps to protect Web applications, Web sites, databases, and any other low hanging fruit that is so tempting to cyber criminals.

Pattern recognition: makes use of a rule set to detect patterns that indicate a possible attack. If an attack is detected, this engine deals with the attack according to how dotDefender is configured. Attacks that pattern recognition works to defend against include:

- Cross-Site Scripting
- SQL Injection
- Path Traversal
- Remote Command Execution
- Probes

- Header Tampering
- Encoding

Session protection: focuses directly on the user session to deal with spoofing and flooding the server with HTTP requests. The session protection engine helps you protect your Web site from:

- Session Hijacking
- Denial of Service Attacks
- Cookie Tampering

Signature knowledgebase: dotDefender's engine uses signatures to detect known attacks, such as vulnerability scanners, bots, site-scrapers, email harvesters, and leeches. As a result, your Web site is protected against:

- Spammer Bots
- Worms
- Bad User Agents
- Compromised Servers

Data leakage protection: prevent sensitive information disclosure using built-in and extensible outgoing traffic inspection rules. Mitigate proliferation of credit card, personal information, application error messages into the wrong hands.

Upload inspection: upload content inspection enforces file extension and MIME-type filtering. Prevent Web shells, backdoors and rootkits from being uploaded via Web content management systems. Scan contents of uploaded files to ensure malicious payloads are not smuggled in posing as benign pictures and content.

Benefits of dotDefender

dotDefender delivers an out-out-of-the-box security solution that can be easily installed with a few simple clicks. Whether an organization has a dedicated IT security department or relies on individuals to take on security responsibilities as part of their daily duties, dotDefender is the ideal choice. Once deployed, dotDefender immediately begins protecting Web applications from attack using its default installation, or it can be customized to the unique needs of any type of company or organization.

As budgets are continuously scrutinized, security is one area that often finds itself in danger of potential cuts. Decision makers are pleased to find that dotDefender delivers the best Total Cost of Ownership (TCO) in the industry providing the best value for each of these elements:

Low cost of acquisition: dotDefender is an affordable solution with several pricing models including SaaS, perpetual and enterprise licenses. License only what you need with no investment in excess capacity or high availability solutions.

Low cost of implementation: dotDefender is a plug & play software solution. With its predefined out-of-the-box Web application security profiles, initial implementation is immediate and simple. No Web application security skills are required to configure and deploy dotDefender.

Low cost of maintenance: maintenance is the most expensive component in the TCO of Web application security technology. dotDefender is application-agnostic, so any change in the application is transparent to the security configuration. dotDefender automatically detects and blocks attack attempts, logs the information, and generates reports and alerts. Automatic updates against emerging threats ensure that your Website is always protected. Multi-platform support for all servers and central management for control and reporting further facilitates and reduces your maintenance efforts.

Finally, dotDefender protects your Web applications. Period. Deploying dotDefender offers your organization piece of mind by:

Stopping major threats at the gateway. Common threats like Injection Attacks, Cross-Site Scripting, Path Traversal, and Remote Command Execution are identified by dotDefender and stopped before they can be used to compromise your systems and data. In addition to these known threats, dotDefender identifies zero-day threats as well by analyzing Website requests for anomalies commonly found in malicious traffic.

Providing PCI Compliance. Simply deploying dotDefender meets the requirement of PCI DSS Section 6.6 without the need for expensive code reviews. dotDefender provides any e-commerce Website an easy, cost effective solution to achieving PCI compliance.

Protection against data leakage. Rules can be specified to filter outgoing traffic as well as incoming requests. This enables better data leakage prevention in addition to improved infrastructure masking. System configuration information, errors, credit card data, and social security numbers are some examples of the data you can protect from accidental exposure.

Web application vulnerabilities will continue to threaten businesses as long as there is a profit to be made by exploiting them. As more laws, regulations, and compliance requirements are put into place to protect data, organizations will be forced to address Web application security or pay a heavy price. Organizations who seek to effectively protect their Websites and data from attack while enjoying a low Total Cost of Ownership will find dotDefender a compelling solution.