SECURITY MODEL FOR WEB APPLICATION SERVICES WITH AWS

R. KIRUTHIKA[1], S. VELRAM[2], A. SUDHARSAN[3], K. RAMESH[4]
1 ASSISTANT PROFESSOR, 2,3,4 UG STUDENTS,
DEPARTMENT OF COMPUTER SCIENCE,
MAHENDRA ENGINEERING COLLEGE, TAMILNADU, INDIA.

ABSTRACT

Web applications are prone to security attacks. Web security is securing a web application layer from attacks by unauthorized users. A lot of the issues that occur over a web application is mainly due to the improper input provided by the client. This we discuss the different aspects of web security and its weakness. The main elements of web security techniques such as the passwords, encryption, authentication and integrity are also discussed. The anatomy of a web application attack and the attack techniques are also covered in details. Explores a number of methods for combatting this class of threats and assesses why they have not proven more successful. This project proposes a better way for minimizing these type of web vulnerabilities. It also provides the best security mechanisms for the said attacks. Web applications are active websites which are composition of server based programs serving user interaction and various other functionalities. Web Server security is thus an important aspect for any organisation having web server connectivity with the internet and also to ensure customers using their websites, for a secure online portal. In this age of digital revolution, there has been a rise in demand of web developers who can produce user friendly web platforms such as mobile applications, web applications. The user base for online web applications is on a rise too. We have seen a huge emphasis on creating visual and catchy web applications but with large amount of sensitive user data at stake there should be more focus on providing web security to the applications developed.

Keywords : Security attacks, Web server & attack techniques.

1. INTRODUCTION

1.1 WEB APPLICATION

Web application security is the process of protecting websites and online services against different security threats that exploit vulnerabilities in an application's code. Common targets for web application attacks are content management systems (e.g., WordPress), database administration tools (e.g., phpMyAdmin) and SaaS applications. The inherent complexity of their source code, which increases the likelihood of unattended vulnerabilities and malicious code manipulation. High value rewards, including sensitive private data collected from successful source code manipulation. Ease of execution, as most attacks can be easily automated and launched indiscriminately against thousands, or even tens or hundreds of thousands of targets at a time. Organizations failing to secure their web applications run the risk of being attacked. Among other consequences, this can result in information theft, damaged client relationships, revoked licenses and legal proceedings.

1.2 WEB APPLICATION VULNERABILITIES

Web application vulnerabilities involve a system flaw or weakness in a web-based application. They have been around for years, largely due to not validating or sanitizing form inputs, misconfigured web servers, and application design flaws, and they can be exploited to compromise the application's security. These vulnerabilities are not the same as other common types of vulnerabilities, such as network or asset. They arise because web applications need to interact with multiple users across multiple networks, and that level of accessibility is easily taken advantage of by hackers.

There are web application security solutions designed specifically for applications, and as such it's important to look beyond traditional vulnerability scanners when it comes to identifying gaps in an organization's application security. To really understand your risks, learn more about some types of web application and cybersecurity attacks.

1.3 AMAZON WEB SERVICE

Amazon web service is a platform that offers flexible, reliable, scalable, easy-to-use and cost-effective cloud computing solutions. AWS is a comprehensive, easy to use computing platform offered Amazon. The platform is developed with a combination of infrastructure as a service (IaaS), platform as a service (PaaS) and packaged software as a service (SaaS) offerings. Amazon Web Services offers a wide range of different business purpose global cloud-based products. The products include storage, databases, analytics, networking, mobile, development tools, enterprise applications, with a pay-as-you-go pricing model.

2. SYSTEM ANALYSIS

2.1 EXISTING SYSTEM

The contraption of this is to propose a new security model focused in access control, cryptographic, cookies and session managements form viewpoint of developers, care for security from an early stage of project development life cycle, used multifactor authentication to improve web applications and service protection. Finally proposes an approach to evaluate and measure security according to software quality standard ISO 25010. The current study will seek to develop simple, efficient and applicable model. Additionally, provide tips for defence programing as security concepts that must be applied with each web application and service. these features are proposed to support the current security models, tools, and techniques.

2.2 PROPOSED SYSTEM

We are using an amazon web services for serverless aplication cloud storage and security purpose. Aws is a cost-effective service that allows you to pay only for what you use, without any up-front or long-term commitments. With aws containing of IAM user authentication and secure using SSM, S3, etc. it's gives server side protection and client side protection. Using virtual private network for authentication. The Motivation of this study is trying to find solutions for access control weaknesses, cookies and session management issues, authentication.

2.3 FEASIBILITY STUDY

To support data collection, a web application called AWS has been developed using PYTHON and MySQL as a database server that fulfills specifications coming from domain experts. Using AWS through home Web access, each patient can access using a personal username and password and compile a daily report of the affecting toxicities by choosing and grading any of them from the user interface. If too much time has passed since the last patient report, AWS will send the doctors a communication about the missing data. This decision also takes into account suggestions about reminders and clinician feedback discussed. Physicians access to a specific part of the site where they can perform managing operations on patients' data, as inserting or updating database information and visualize flowsheets of patient toxicities by means of graphing functions.

3. WEB APPLICATION VULNERABILITIES

3.1 SQL INJECTION

SQL Injection (SQLi) is a type of an injection attack that makes it possible to execute malicious SQL statements. These statements control a database server behind a web application. Attackers can use SQL Injection vulnerabilities to bypass application security measures. They can go around authentication and authorization of a web page or web application and retrieve the content of the entire SQL database. They can

also use SQL Injection to add, modify, and delete records in the database. An SQL Injection vulnerability may affect any website or web application that uses an SQL database such as MySQL, Oracle, SQL Server, or others. Criminals may use it to gain unauthorized access to your sensitive data: customer information, personal data, trade secrets, intellectual property, and more. SQL Injection attacks are one of the oldest, most prevalent, and most dangerous web application vulnerabilities

3.2 CROSS SITE SCRIPTING

Cross-Site Scripting (XSS) attacks are a type of injection, in which malicious scripts are injected into otherwise benign and trusted websites. XSS attacks occur when an attacker uses a web application to send malicious code, generally in the form of a browser side script, to a different end user. Flaws that allow these attacks to succeed are quite widespread and occur anywhere a web application uses input from a user within the output it generates without validating or encoding it. An attacker can use XSS to send a malicious script to an unsuspecting user. The end user's browser has no way to know that the script should not be trusted, and will execute the script. Because it thinks the script came from a trusted source, the malicious script can access any cookies, session tokens, or other sensitive information retained by the browser and used with that site. These scripts can even rewrite the content of the HTML page.

3.3 URL REDIRECTION

URL redirection, also called URL forwarding, is a World Wide Web technique for making a web page available under more than one URL address. When a web browser attempts to open a URL that has been redirected, a page with a different URL is opened. Similarly, domain redirection or domain forwarding is when all pages in a URL domain are redirected to a different domain

URL redirection is done for various reasons:

for URL shortening;

to prevent broken links when web pages are moved;

to allow multiple domain names belonging to the same owner to refer to a single web site;

to guide navigation into and out of a website;

for privacy protection; and

for hostile purposes such as phishing attacks or malware distribution.

3.4 BROKEN AUTHENTICATION

Broken authentication is typically caused by poorly implemented authentication and session management functions. Broken authentication attacks aim to take over one or more accounts giving the attacker the same privileges as the attacked user. Authentication is "broken" when attackers are able to compromise passwords, keys or session tokens, user account information, and other details to assume user identities. Due to poor design and implementation of identity and access controls, the prevalence of broken authentication is widespread. Common risk factors include:

Predictable login credentials

User authentication credentials that are not protected when stored

Session IDs exposed in the URL (e.g., URL rewriting)

Session IDs vulnerable to session fixation attacks

Session value that does not time out or get invalidated after logout

Session IDs that are not rotated after successful login

Passwords, session IDs, and other credentials sent over unencrypted connections.

3.5 DISTRIBUTED DENIAL SERVICE

A Distributed Denial of Service (DDoS) attack is a non-intrusive internet attack made to take down the targeted website or slow it down by flooding the network, server or application with fake traffic. When against a vulnerable resource-intensive endpoint, even a tiny amount of traffic is enough for the attack to succeed. Distributed Denial of Service (DDoS) attacks are threats that website owners must familiarize themselves with as they are a critical piece of the security landscape. Navigating the various types of DDoS attacks can be challenging and time consuming. DDoS attack is to prevent legitimate users from accessing your website. For a DDoS attack to be successful, the attacker needs to send more requests than the victim server can handle. Another way successful attacks occur is when the attacker sends bogus requests.

3.6 SERVER SIDE REQUEST FORGERY

Server-side request forgery (also known as SSRF) is a web security vulnerability that allows an attacker to induce the server-side application to make HTTP requests to an arbitrary domain of the attacker's choosing. In typical SSRF examples, the attacker might cause the server to make a connection back to itself, or to other web-based services within the organization's infrastructure, or to external third-party systems. SSRF exploit that causes connections to external third-party systems might result in malicious onward attacks that appear to originate from the organization hosting the vulnerable application, leading to potential legal liabilities and reputational damage.

3.7 SERVER SIDE REQUEST FORGERY(SSH)

Server-Side Request Forgery allows an attacker to make local and/or remote network requests while masquerading as the target server.

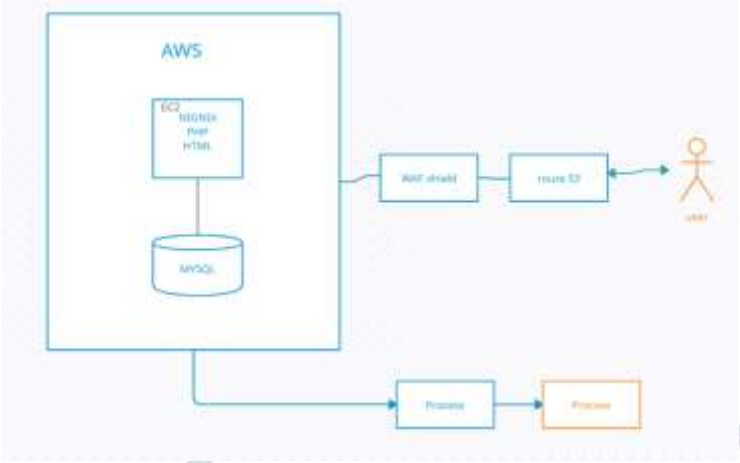
Having an SSH endpoint that is accessible through SSRF may lead to total compromise of the target computer other resources that are accessible by the compromised account.

4. SYSTEM REQURIMENT

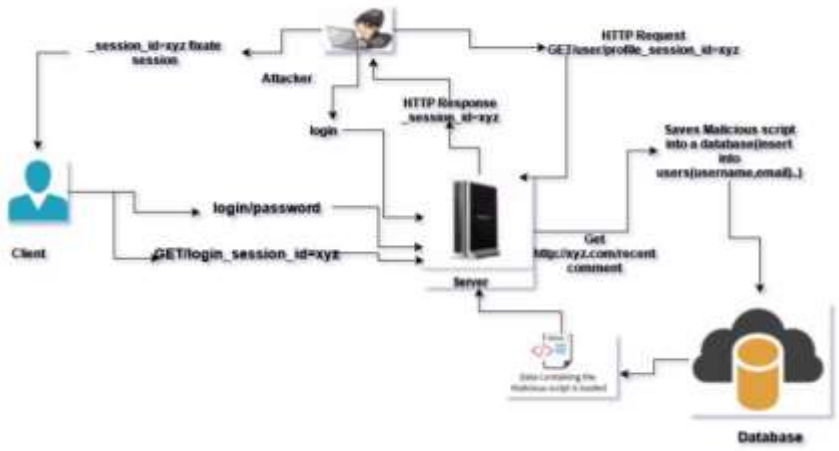
Using AWS, you will gain the control and confidence you need to securely run your business with the most flexible and secure cloud computing environment available today. As an AWS customer, you will benefit from AWS data centers and a network architected to protect your information, identities, applications, and devices. With AWS, you can improve your ability to meet core security and compliance requirements, such as data locality, protection, and confidentiality with our comprehensive services and features. AWS allows you to automate manual security tasks so you can shift your focus to scaling and innovating your business. Plus, you pay only for the services that you use. All customers benefit from AWS being the only commercial cloud that has had its service offerings and associated supply chain vetted and accepted as secure enough for top-secret workloads.

4.1 USE CASE DAIGRAM

Use case diagram consists ec2 linux and software configured in the linux by linux command and data are storedin the mysql and the are portected by the cloud watch and waf,route 53 and used IAM ascces admin.



4.2 SYSTEM ARCHITECTURE



In our architecture we describe the all sessions. Client: The client of a web browser is effectively making client requests for pages from servers all over the web. In this Attack Goal % Stealing Sensitive Information 42% Defacement 23% Planning Malware 15% Unknown 08% Deceit 03% Blackmail 02% Link Spam 03% Worm 01% Phishing 01% Information Warfare 01% 2017 6th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), Sep. 20-22, 2017, AIIT, Amity University Uttar Pradesh, Noida, India 453 article client login to system normally, client sends request to server and gets response. This happens only in normal scenario.

Attacker: Attacker is an unauthorized user. Typically, this kind of attacker would be a proficient programmer or engineer with sufficient technical knowledge to understand the weak points in a security system. In this article Attacker attacks the website through SQL injection and XSS. Uses of SQL injection and XSS by the attacker is mentioned below.

4.3 DATA FLOW DAIGRAM

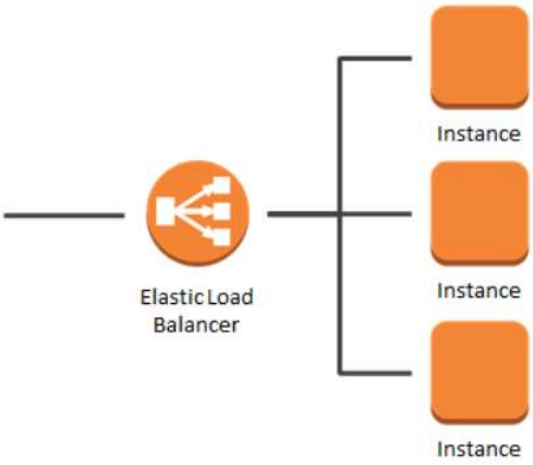
EC2

Amazon Elastic Compute Cloud (Amazon EC2) provides scalable computing capacity in the Amazon Web Services (AWS) Cloud. Using Amazon EC2 eliminates your need to invest in hardware up front, so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. Amazon EC2 enables you to scale up or down to handle changes in requirements or spikes in popularity, reducing your need to forecast traffic.

CLOUD WATCH

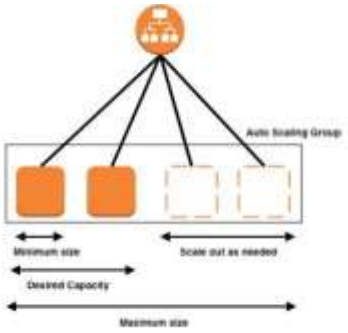
CloudWatch provides you with data and actionable insights to monitor your applications, respond to system-wide performance changes, optimize resource utilization, and get a unified view of operational health. CloudWatch collects monitoring and operational data in the form of logs, metrics, and events, providing you with a unified view of AWS resources, applications, and services that run on AWS and on-premises servers. You can use CloudWatch to detect anomalous behavior in your environments, set alarms, visualize logs and metrics side by side, take automated actions, troubleshoot issues, and discover insights to keep your applications running smoothly.

LOAD BALNCER



A load balancer distributes the incoming application or network traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses, in multiple Availability Zones. It uses health checks to detect which instances are healthy and directs traffic only across those instances.

AUTOSCALING

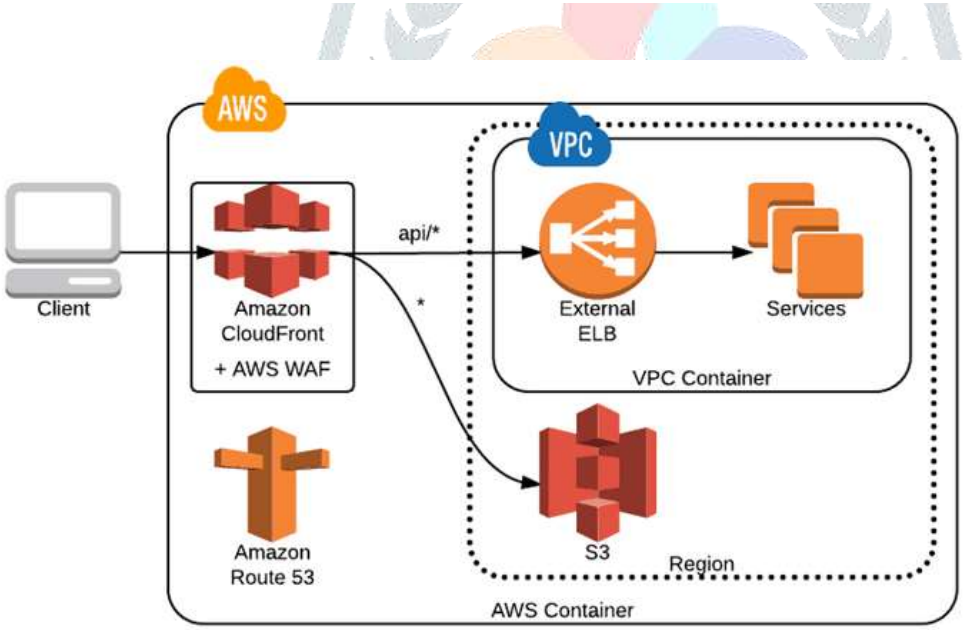


Auto Scaling monitors your application & automatically adjusts the capacity to maintain steady, Predictable performance at the lowest possible cost.

EASTIC IP

An *Elastic IP address* is a static, public IPv4 address designed for dynamic cloud computing. You can associate an Elastic IP address with any instance or network interface in any VPC in your account. With an Elastic IP address, you can mask the failure of an instance by rapidly remapping the address to another instance in your VPC.

5. CONTROL FLOW DAIGRAM



CLOUD FRONT

Amazon Cloud Front is a fast content delivery network (CDN) service that securely delivers data, videos, applications, and APIs to customers globally with low latency, high transfer speeds, all within a developer-friendly environment.

WAF

AWS WAF is a web application firewall that lets you monitor the HTTP and HTTPS requests that are forwarded to CloudFront, and lets you control access to your content. Based on conditions that you

specify, such as the values of query strings or the IP addresses that requests originate from, CloudFront responds to requests either with the requested content or with an HTTP status code 403 (Forbidden). You can also configure CloudFront to return a custom error page when a request is blocked

S3

The S3 in Amazon S3 stands for **Simple Storage Service**. As the name implies it is a web service provided by Amazon Web Services which provides storage for the internet. This storage is **highly-scalable and secure in the cloud**. Having data stored in the cloud eliminates the need for in-house storage and customers can opt for unlimited storage or buy more as it is needed. S3 is an incredibly helpful product which allows users to store and retrieve data from anywhere on the web, at any time. This is done though the AWS Management Console which is an easy to use web interface.

5.1 PREVENTION AND SECURITY

SSM

AWS Systems Manager Agent (SSM Agent) is Amazon software that can be installed and configured on an EC2 instance, an on-premises server, or a virtual machine (VM). The agent processes request from the Systems Manager service in the AWS Cloud, and then runs them as specified in the request. AWS service that you can use to view and control your infrastructure on AWS. Using the Systems Manager console, you can view operational data from multiple AWS services and automate operational tasks across your AWS resources.

HTTPS & SSL

SSL stands for Secure Sockets Layer and, in short, it's the standard technology for keeping an internet connection secure and safeguarding any sensitive data that is being sent between two systems, preventing criminals from reading and modifying any information transferred, including potential personal details. The two systems can be a server and a client (for example, a shopping website and browser) or server to server (for example, an application with personal identifiable information or with payroll information). It does this by making sure that any data transferred between users and sites, or between two systems remain impossible to read. It uses encryption algorithms to scramble data in transit, preventing hackers from reading it as it is sent over the connection. This information could be anything sensitive or personal which can include credit card numbers and other financial information, names and addresses. HTTPS (Hyper Text Transfer Protocol Secure) appears in the URL when a website is secured by an SSL certificate. The details of the certificate, including the issuing authority and the corporate name of the website owner, can be viewed by clicking on the lock symbol on the browser bar.

IAM

You can use AWS IAM to securely control individual and group access to your AWS resources. You can create and manage user identities ("IAM users") and grant permissions for those IAM users to access your resources. You can also grant permissions for users outside of AWS (federated users). IAM makes it easy to provide multiple users secure access to your AWS resources. IAM enables you to:

Manage IAM users and their access: You can create users in AWS's identity management system, assign users individual security credentials (such as access keys, passwords, multi-factor authentication devices), or request temporary security credentials to provide users access to AWS services and resources. You can specify permissions to control which operations a user can perform.

Manage access for federated users: You can request security credentials with configurable expirations for users who you manage in your corporate directory, allowing you to provide your employees and applications secure access to resources in your AWS account without creating an IAM user account for them. You specify the permissions for these security credentials to control which operations a user can perform.

CLOUD WATCH

Security of the cloud – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs. To learn about the compliance programs that apply to Amazon Work Spaces, see AWS Services in Scope by Compliance Program.

Security in the cloud – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations

6. REFERENCE

[1]https://aws.amazon.com/step-functions/?step-functions.sort-by=item.additionalFields.postDateTime&step-functions.sort-order=desc
[2] https://aws.amazon.com/getting-started/fundamentals-core-concepts/
[3] https://ieeexplore.ieee.org/document/8342469
[4] https://ieeexplore.ieee.org/document/7176244
5)https://ieeexplore.ieee.org/document/8400266
6)https://www.rapid7.com/fundamentals/web-application-security-testing/