Một số kỹ thuật

# Bảo mật website

1. Validate input
   1. Biểu thức chính quy
   2. Isset()
   3. Strlen (‘$\_POST[‘….’])>2
   4. Filter\_var()
   5. Filter\_validate\_email, filter\_validate\_int, filter\_sanitize\_string, sanitize\_special\_chars
   6. Strip\_tags
2. Escape output: Thứ tự mức độ mạnh của các hàm:
   1. Htmlspecialchars()
   2. Htmlentities()
      1. VD: echo htmlentities($\_POST[‘text’]);
      2. echo htmlentities($\_POST[‘text’],’ENT\_COMPAT’,’utf-8’);
   3. Strip\_tags($var, ‘<p><strong>’)
3. SQL injection

Luôn dùng mysqli\_real\_escape…

1. Session hack;
   1. Không cho sử dụng phpssid var trên url bằng cách khai báo: init\_set(‘session\_use\_only\_cookies’, true); 🡪 hạn chế được việc không cho sử dụng phpssid thông qua url.

VD: ?phpssid=xxxxxxxx

Vì việc sử dụng theo url có thể người dùng dễ chiếm quyền điều khiển admin.

* 1. Sử dụng hàm: Session\_regenerate\_id()

Khi người dùng đăng nhập, sau bước kiểm tra user và pass, nếu ok thì tạo lại id session mới trước khi lưu thông tin đăng nhập vào session

* 1. Nếu muốn an toàn hơn nữa thì giới hạn thời gian session khoảng 30s thì tạo lại id session . VD:  
     if(!isset($\_SESSION[‘time’]) || $\_SESSION[‘time’] < (time()-30)){

session\_regenerate\_id();

$\_SESSION[‘time’] = time();

}

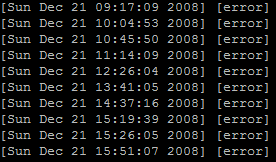
5. Sử dụng hợp lý Error Reporting

Trong quá trình phát triển, cảnh báo lỗi là công cụ tốt nhất của bạn. Các cảnh báo lỗi này giúp bạn tìm ra những lỗi chính tả trong biến, phát hiện hàm sử dụng sai và nhiều hơn thế. Tuy nhiên, khi trang web của bạn đã được trực tuyến thì các thông báo lỗi lại trở thành “kẻ thù” vì nó có thể cho người dùng biết rất nhiều thông tin về trang web (phần mềm bạn sử dụng, cấu trúc folder…).

Khi trang web đã đi vào hoạt động, bạn nên đảm bảo rằng đã ẩn tất cả các thông báo lỗi. Điều đó có thể thực hiện bằng cách sử dụng hàm đơn giản sau:

|  |  |
| --- | --- |
|  | error\_reporting(0); |
|  |  |

Nếu chức năng nào đó hoạt động không đúng, bạn vẫn muốn và cần biết về nó. Vì vậy bạn nên luôn đảm bảo rằng bạn đã log lại lỗi xảy ra vào một file được bảo vệ với hàm [set\_error\_handler](http://php.net/set_error_handler).



6. Vô hiệu hóa những “tính năng yếu” của PHP

Từ những ngày đầu tiên, nhà thiết kế PHP luôn thêm vào những tính năng giúp việc phát triển dễ dàng hơn. Một vài tính năng hữu ích có thể gây ra các hệ quả không lường trước được. Những tính năng này cho phép kiểm duyệt dữ liệu trở nên khó khăn và tạo ra những lỗi hệ thống. Một trong những điều đầu tiên bạn nên làm khi bắt đầu phát triển là vô hiệu hóa những tính năng này.

Lưu ý: Việc vô hiệu hóa “tính năng yếu” hay không phụ thuộc vào host của bạn. Nếu bạn phát triển trên máy tính riêng hay các những môi trường cục bộ tương tự, chúng có thể bắt buộc phải bật. Một vài tính năng sẽ được loại bỏ trong bản PHP7.

Register Globals (register\_globals)

register\_globals giúp tăng tốc việc phát triển ứng dụng.

VD:URL: http://yoursite.tld/index.php?var=1 bao gồm một truy vấn string. register\_globals cho phép chúng ta truy cập vào giá trị với $var thay vì $\_GET[‘var’] một cách tự động. Điều này nghe có vẻ hữu ích với bạn nhưng không may tất cả các biến trong code đều có thuộc tính này và chúng ta có thể dễ dàng truy cập vào ứng dụng PHP không có bảo vệ. Đoạn code dưới đây là một ví dụ mà bạn thường thấy trong PHP script:

if( !empty( $\_POST['username'] ) && $\_POST['username'] == 'test' && !empty( $\_POST['password'] ) && $\_POST['password'] == "test123" )

{

    $access = true;

}

Thật không may chúng ta không thể vô hiệu hóa register\_globals từ phía script nhưng chúng ta có thể sử dụng tệp tin .htaccess để làm điều này. Một vài host cho phép bạn chứa file php.ini trên máy chủ. Nếu ứng dụng sử dụng register\_globals, người dùng sẽ có thể đặt access=1 vào một truy vấn string và sẽ có thể truy cập vào bất cứ script nào đang chạy.

Vô hiệu hóa với .htaccess

php\_flag register\_globals 0

Vô hiệu hóa với php.ini

register\_globals = Off

Lưu ý: Nếu bạn sử dụng file php.ini không phù hợp với toàn bộ máy chủ, bạn cần bổ sung khai báo ở mỗi thư mục phụ có PHP.



Magic Quotes (magic\_quotes\_gpc, magic\_quotes\_runtime, magic\_quotes\_sybase)

Magic Quotes là một tính năng giúp lập trình viên tránh rắc rối khi sử dụng addslashes() và những tính năng bảo mật tương tự trong code. Có ít nhất 3 vấn đề với Magic Quotes. Một vấn đề xảy ra khi cả Magic Quotes và addslashes() đều được sử dụng. Trong trường hợp này, nếu bạn có thể kết thúc chuỗi với nhiều dấu sổ/ có thể sẽ gây ra lỗi. Vấn đề thứ hai là nếu bạn bật Magic Quotes nhưng nó không hoạt động. Điều này khiến tất cả dữ liệu đầu vào đều không được kiểm tra. Vấn đề thứ ba là Magic Quotes sử dụng dấu sổ đơn hoặc kép nhưng nếu bạn sử dụng một database engine, có rất nhiều kí tự đặc biệt cần được sử dụng. Vì vậy bạn nên vô hiệu hóa tính năng này và sử dụng một biến phù hợp.

Vô hiệu hóa với .htaccess

php\_flag magic\_quotes\_gpc 0 php\_flag magic\_quotes\_runtime 0

Vô hiệu hóa với php.ini

magic\_quotes\_gpc = Off

magic\_quotes\_runtime = Off

magic\_quotes\_sybase = Off

Lưu ý: Nếu bạn sử dụng file php.ini không phù hợp với toàn bộ máy chủ, bạn cần bổ sung khai báo ở mỗi thư mục phụ có PHP.

# Tăng tốc độ website

1. Sử dụng các template giao diện đơn giản

2. Sử dụng cache là điều bắt buộc

3. Sử dụng host có máy chủ ở gần bạn

4. Hãy sử dụng server riêng nếu có thể (VPS/Dedicated Server)

5. Sử dụng CDN

CDN (Content Devilery Network) là một công nghệ truyền tải các dữ liệu tĩnh (hình ảnh, file CSS và Javascript,..) thông qua proxy của các máy chủ CDN, nó sẽ tự nhận diện địa chỉ của người dùng và sẽ điều hướng họ qua proxy ở server gần nhất mà nó có hỗ trợ. CDN cực kỳ hữu dụng nếu bạn sử dụng các host nước ngoài và có lượng lớn hình ảnh trên website.

6. Tối ưu hình ảnh

7. Tối ưu database

Còn nếu website bạn có mức độ lớn hơn, bạn có điều kiện hơn thì hãy sử dụng một máy chủ riêng để lưu trữ và xử lý database (Remote MySQL Server) vì việc xử lý database khá tốn tài nguyên nên nó sẽ làm ảnh hưởng đến việc xử lý các file PHP của server. Chỉ đơn giản là tìm một cái host khác và bỏ database vào đó rồi sửa phần hostname trong file wp-config.php thành địa chỉ của máy chủ đó. Hoặc bạn cũng có thể dùng các dịch vụ chuyên dụng hơn như Amazon RDS, Google Cloud Database.

8. Hạn chế kết nối với tài nguyên bên ngoài

9. Gzip và Cache cho trình duyệt

Để kích hoạt gzip bạn có thể chèn đoạn sau vào file .htaccess (nếu sử dụng Shared Host/Apache):

# Start gzip compression

AddOutputFilterByType DEFLATE text/plain

AddOutputFilterByType DEFLATE text/html

AddOutputFilterByType DEFLATE text/xml

AddOutputFilterByType DEFLATE text/css

AddOutputFilterByType DEFLATE application/xml

AddOutputFilterByType DEFLATE application/xhtml+xml

AddOutputFilterByType DEFLATE application/rss+xml

AddOutputFilterByType DEFLATE application/javascript

AddOutputFilterByType DEFLATE application/x-javascript

# End Gzip compression

Còn bạn dùng NGINX thì hãy chắc chắn file nginx.conf của bạn có các dòng này:

# Start Gzip compression

gzip on;

gzip\_min\_length 1100;

gzip\_buffers 4 32k;

gzip\_types text/plain application/x-javascript text/xml text/css;

gzip\_vary on;

# End Gzip Compression

Kết quả kiểm tra tại trang http://checkgzipcompression.com/ có thể cho bạn thấy website của bạn đã bật gzip chưa và đã được nén xuống còn bao nhiêu.

Để thiết lập cache trình duyệt, bạn có thể chèn đoạn dưới vào file .htaccess (Apache):

# Browser caching

<IfModule mod\_expires.c>

ExpiresActive On

ExpiresByType image/jpg "access 1 year"

ExpiresByType image/jpeg "access 1 year"

ExpiresByType image/gif "access 1 year"

ExpiresByType image/png "access 1 year"

ExpiresByType text/css "access 1 month"

ExpiresByType application/pdf "access 1 month"

ExpiresByType application/x-javascript "access 1 month"

ExpiresByType application/x-shockwave-flash "access 1 month"

ExpiresByType image/x-icon "access 1 year"

ExpiresDefault "access 2 days"

</IfModule>

# End Browser caching

Hoặc cho NGINX:

location ~\* .(jpg|jpeg|png|gif|ico|css|js)$ {

expires max;

}

10. Sử dụng Host có ổ cứng SSD