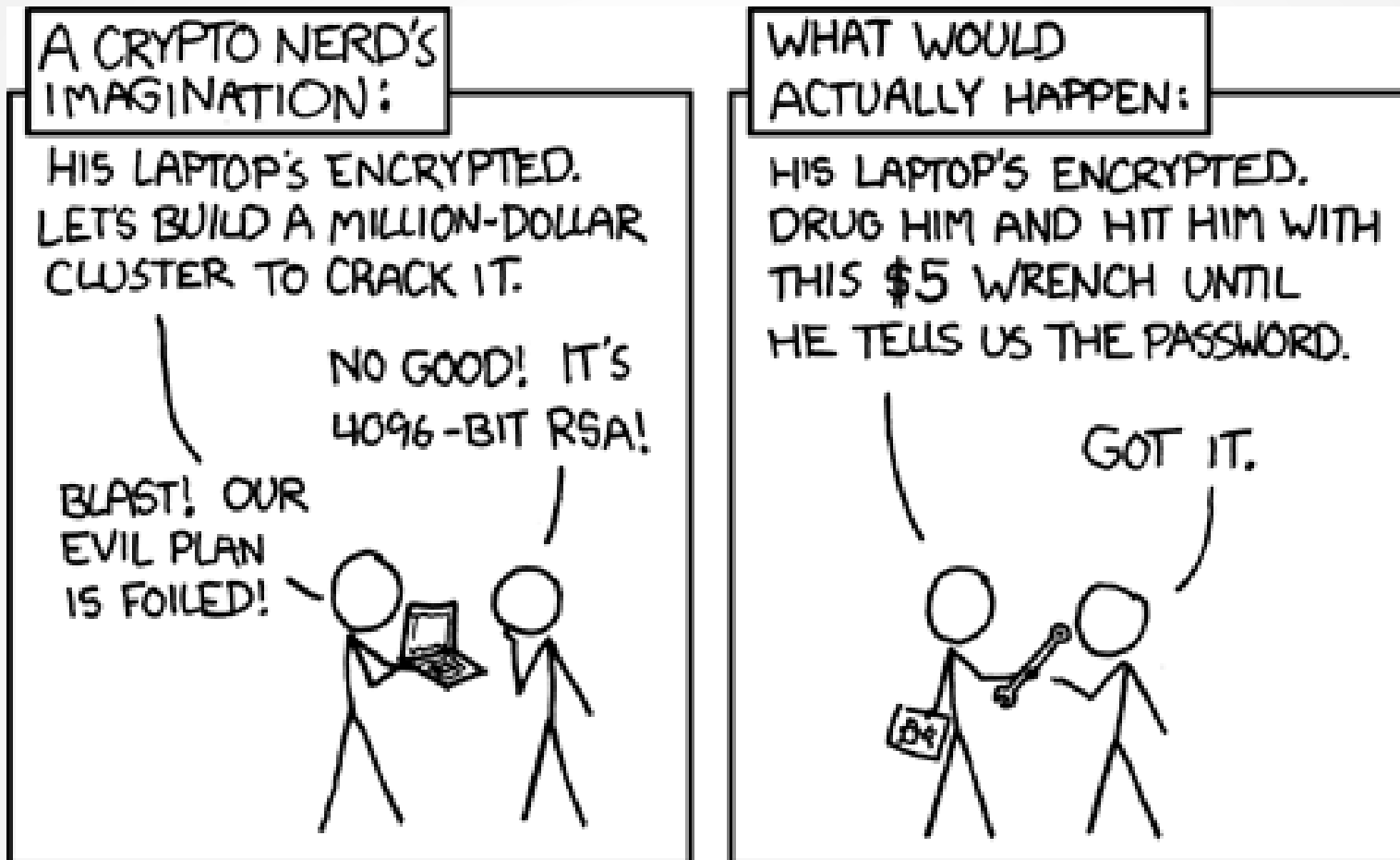



Sicherheit in Web-Apps

„Sicherheit“



HTTPS

(theoretischer) Schutz gegen Man-in-the-middle-Angriffe:



This is probably not the site that you are looking for!

You attempted to reach **forums.mcafeehelp.com**, but instead you actually reached a server identifying itself as **community.mcafee.com**. This may be caused by a misconfiguration on the server or by something more serious. An attacker on your network could be trying to get you to visit a fake (and potentially harmful) version of **forums.mcafeehelp.com**. You should not proceed.

▶ [Help me understand](#)

What could possibly go wrong?

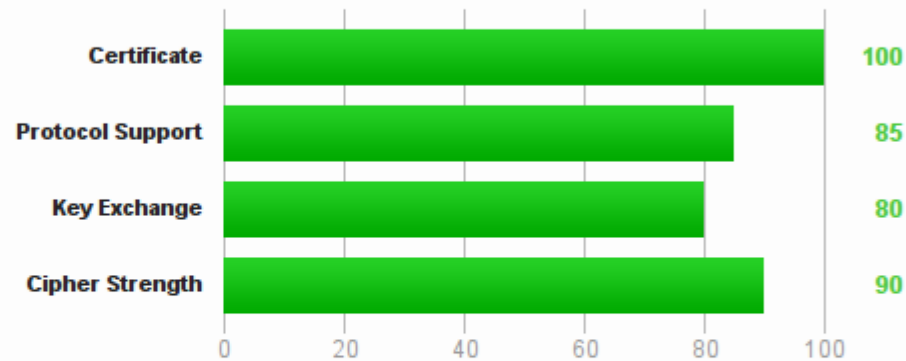
HTTPS (cont.)

- Attacken: BEAST/CRIME, sslsniff, sslstrip
- Komplette Verschlüsselung, nicht nur Login!
- (Session-)Cookies absichern:
 - Set-Cookie: [...]; **secure**
 - Django: **SESSION_COOKIE_SECURE**
- HTTP Strict Transport Security:
 - Strict-Transport-Security: max-age=expireTime
 - Erster Request weiterhin gefährdet

HTTPS (cont.)

Summary

Overall Rating



Documentation: [SSL/TLS Deployment Best Practices](#) and [SSL Server Rating Guide \[Updated\]](#).

This server is vulnerable to the [CRIME attack](#). Grade capped to B.

This server is vulnerable to the [BEAST attack](#). Grade capped to B.

<https://www.ssllabs.com/ssltest/index.html>

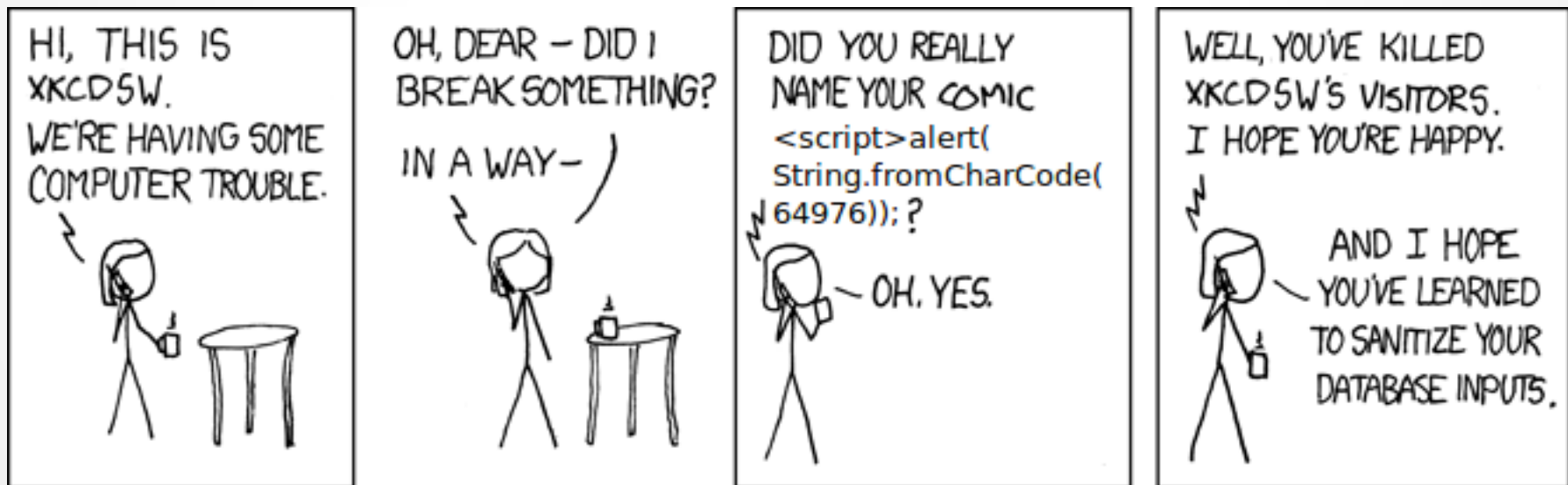
Cross-site scripting (XSS)

Ausführen von beliebigem Code (html/css/js) im Securitycontext der angegriffenen Seite

```
@app.route('/search')
```

```
def search():
```

```
    return 'You searched for: %s' % request.args['q']
```



XSS (cont.)

- Userinput ist nicht vertrauenswürdig
- Speicherung der Daten as is
- Contextspezifisches Escaping bei der Ausgabe
- TRACE method deaktivieren (XST)
- Zugriff auf Cookies aus JS verbieten:
 - Set-Cookie: [...]; httpOnly
 - Django: SESSION_COOKIE_HTTPONLY
- Content Security Policy:
 - Content-Security-Policy: default-src 'none'; ↵
 - script-src https://cdn.mybank.net;

CSRF

Ein Formular genügt auf der Seite des Angreifers genügt:

```
<form action="http://paste.ubuntuusers.de/" method="post">  
  <div style="display:none;">  
    <input name="title" value="OWNED"/>  
    <input name="code" value="foobar"/>  
  </div>  
  <input type="submit" value="Show pics"/>  
</form>
```

Bonus: Autosubmit via Javascript

CSRF (cont)

- GET darf keine Ressourcen verändern!
- Alles andere muss mit CSRF-Tokens gesichert sein
- Django:

`django.middleware.csrf.CsrfViewMiddleware`

`{% csrf_token %}`

```
<form action="" method="post" id="login-form">  
  <input type='hidden' name='csrfmiddlewaretoken'  
    value='random_eg_42' />
```

Clickjacking

Harmlos?



It's easy to customize your Firefox exactly the way you want it.
[Choose from thousands of add-ons.](#)



Clickjacking (Cont.)

Nicht ganz :)



The screenshot shows a news article on the website heise.de. The article title is "Alert! Großes Notfall-Update für Java". Below the title, there are options for "vorlesen / MP3-Download" and a link to the "English Version: Oracle releases emergency patches for Java". Navigation links for "« Vorige | Nächste »" are visible. A watermark of a blue globe with a white mouse cursor is overlaid on the article content. Below the article, there are social media sharing buttons for "Empfehlen" (304), "Tweet" (184), and "+1" (34). The article is part of a "Themen-Forum Desktopsicherheit" with 539 comments.

Alert! Großes Notfall-Update für Java
vorlesen / MP3-Download
English Version: Oracle releases emergency patches for Java
« Vorige | Nächste »
Version zum Drucken | Per E-Mail versenden
Permalink: http://heise.de/-1796504
Kommentare lesen (539 Beiträge) Themen-Forum Desktopsicherheit
Empfehlen 304 Tweet 184 +1 34



It's easy to customize your Firefox exactly the way you want it.
Choose from thousands of add-ons.

Clickjacking (Cont.)

- Framing via Javascript verbieten
- Oder in neuen Browsern: X-Frame-Options
 - X-Frame-Options: DENY
- Django:
 - d.m.clickjacking.XFrameOptionsMiddleware
 - X_FRAME_OPTIONS = 'DENY'

Fragen? & Danke!