

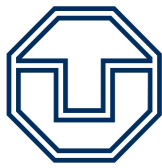


**TECHNISCHE  
UNIVERSITÄT  
DRESDEN**

**Department of Computer Science** Institute of System Architecture, Operating Systems Group

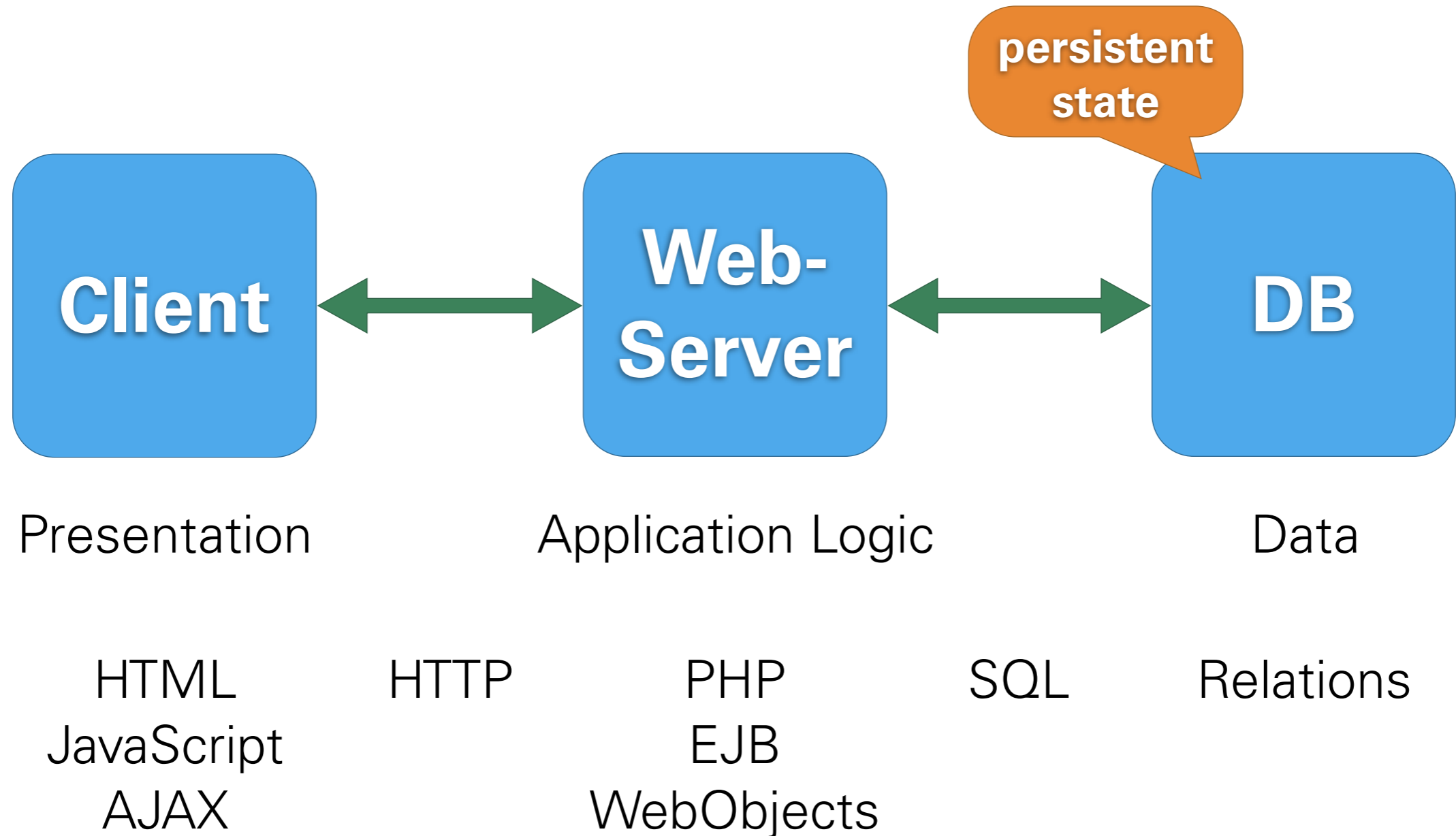
# **PROBLEMS IN PRACTICE: THE WEB**

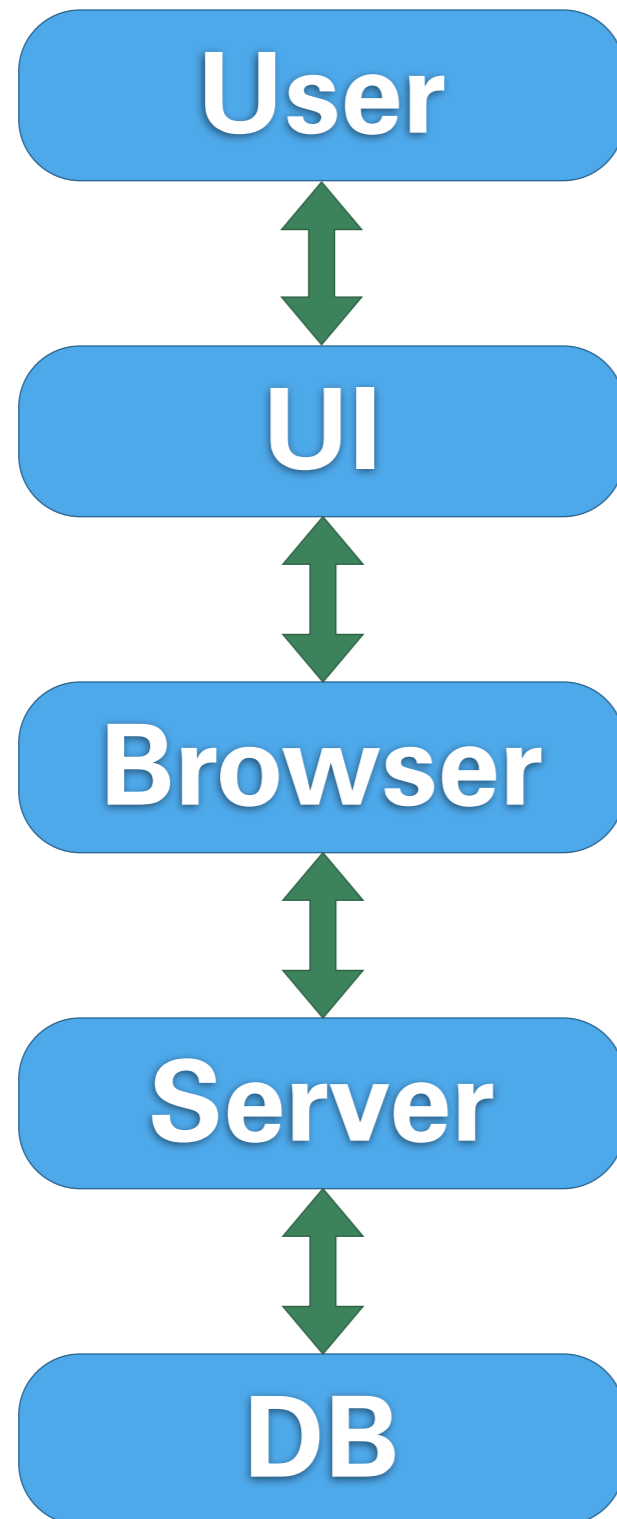
**MICHAEL ROITZSCH**



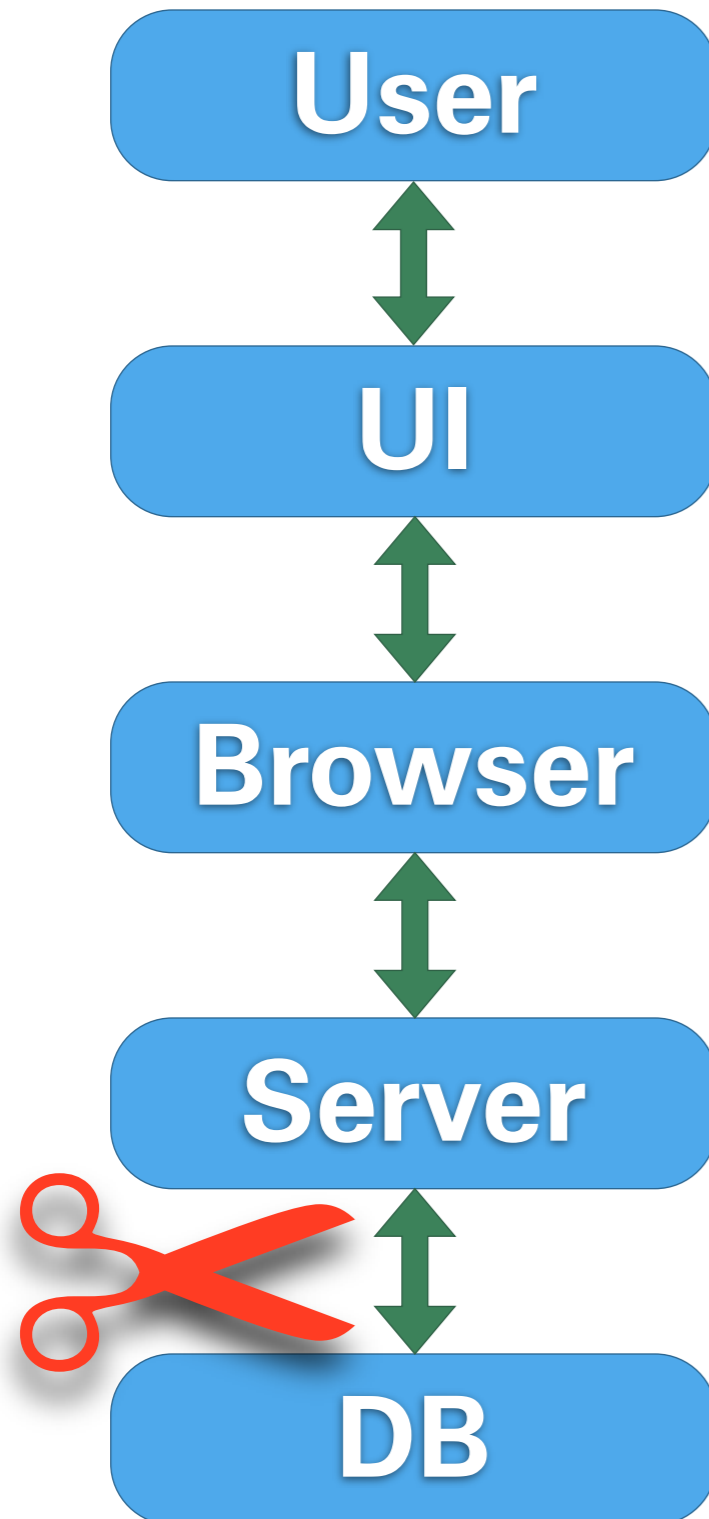
# THE WEB AS A DISTRIBUTED SYSTEM

# WEB HACKING SESSION





- user accesses a sensitive service
- attacker tries to disturb
- various complex layers
- independently developed technologies are being combined
- what you see may not be what you get...



- **goal:** manipulate state stored in the backend DB
- not directly accessible (hopefully)
- improper input checking in frontend server required
- nice: inconsistency is persistent

```
$password = $_POST['password'];
```

```
$id = $_POST['id'];
```

```
$sql = "UPDATE Accounts SET  
PASSWORD = '$password' WHERE  
account_id = $id";
```

Now imagine: `password=' ; --`

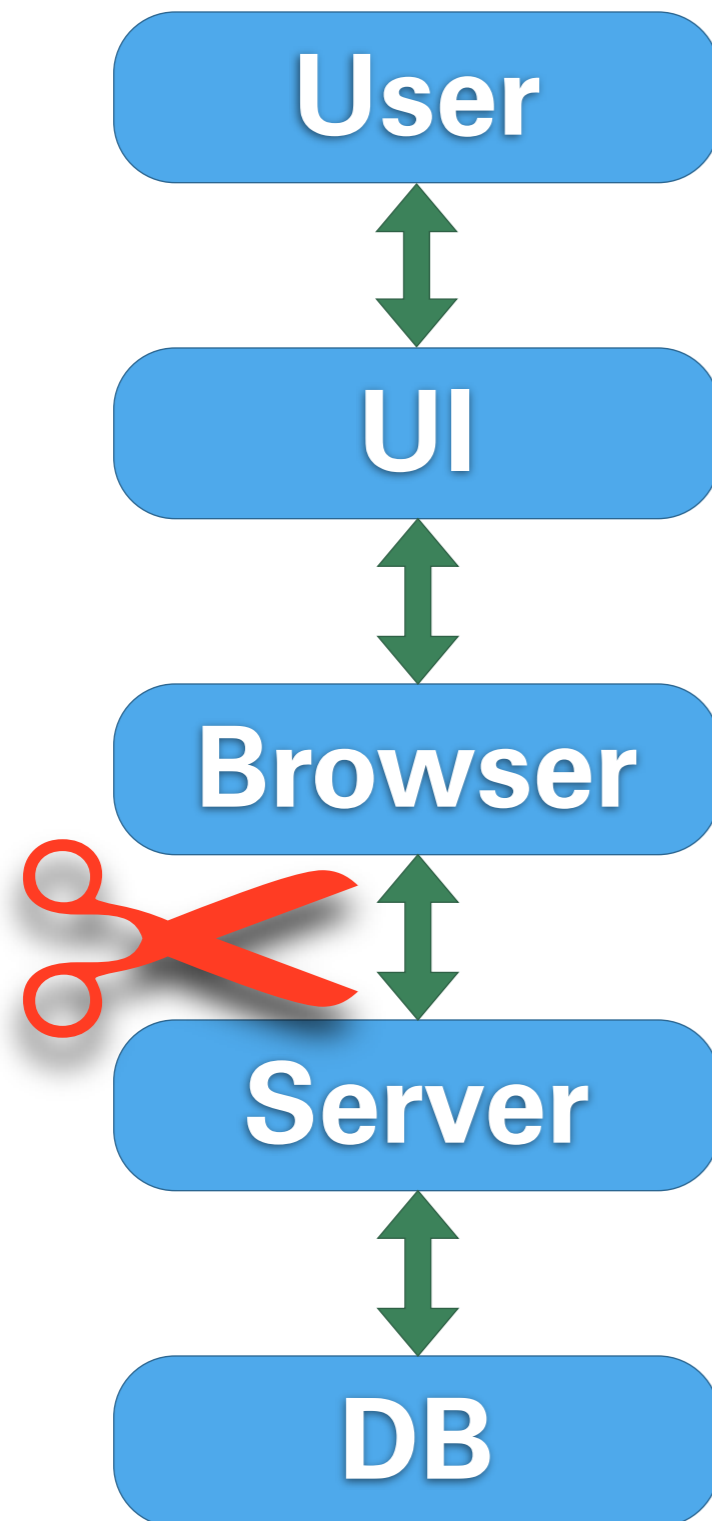
## **SQL injection**



Comic by Randall Munroe, [xkcd.com](http://xkcd.com)







- **goal:** manipulate content delivered to the browser
- infrastructure attacks like DNS cache poisoning
- solution for this: make sure you use SSL
- ... and check CRLs
- improper input checking can still bite you

- `http://example.com/?query=query string`
- generates website containing:  
`<p>You are looking for: query string</p>`
- so how about that:  
`http://example.com/?query=HTML code`
- remember that?  
`http://www.wolfgang-schaeuble.de/?  
search=</strong></div>...`



Dr. Wolfgang Schäuble MdB  
Bundesminister des Innern  
CDU/CSU-Bundestagsfraktion CDU-Deutschlands

24.05.2008

### Bundesinnenminister tritt zurück

wäre eine Meldung, die sicher viele gerne lesen würden. Allerdings handelt es sich nur um eine Cross-Site-Scripting-Schwachstelle im der Webseite des Politikers, der gerne die Online-Durchsuchung einführen möchte. Scherzbolde können dadurch beliebige Meldungen unter der Domäne wolfgang-schaeuble.de erstellen.

Der Fehler liegt in der Suchfunktion des Internetauftritts, die HTML- und Skriptcode in Anfragen nicht ausfiltert. Grüße an dmk.

Suchen...

Position

Verfassungsschutzbericht →

BKA-Gesetz →

Fertig

- Can you steal site credentials with this?
- imagine a bank website allowing injection
- What do we have?
  - there is the standard bank login on the page
  - you can inject a script into the page
  - you want to keep the login form functional
- How do you get the password?

- JavaScript can access password fields
- you cannot use AJAX to send the password
- **same origin policy**
  - JavaScript may only connect back to the originating server (with some tolerance)
- can be defeated with `<img>` tags
  - encode password in URL to ping your server
- JavaScript can also read cookies...

- disallow cross-site image loading?
  - lots of sites use this
- no JavaScript access to password field?
  - AJAX logins need this
- fix web application
  - well...

techcrunch.com

googleadservices.com

facebook.com

snap.com

wordpress.com

topsy.com

doubleclick.net

undertone.com

quantserve.com

crunchboard.com

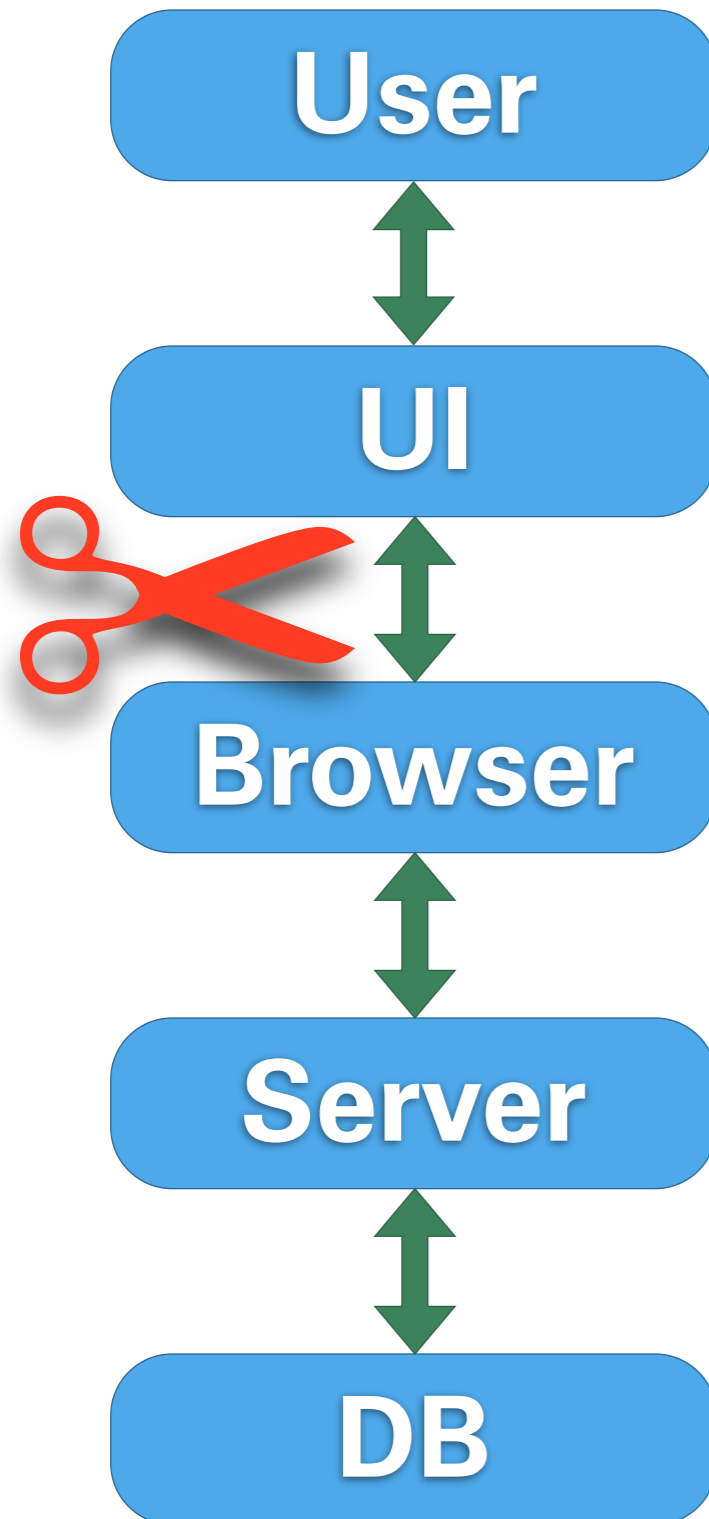
google-analytics.com

scorecardresearch.com

ixnp.com

fbcdn.net





- **goal:** trick the browser to not show what's actually happening
- or: how to pull strings behind the user's back
- or: can one website control another one?
- no mischief with the server communication

- user visits a regular website you control
- Can you obtain credentials of a different site?
- some preconditions
  - user is logged in to the target site in another browser tab
  - the target site identifies the user session with a cookie
- no cross-site cookie leakage in browser

- same origin policy prevents AJAX to target
- again, `<img>` is your friend
- one website can send arbitrary requests to another, unrelated site
- **cross site request forgery**
- a special case of the **confused deputy problem**
- requests are blindly operating the target

- send requests and GET parameters
  - click buttons in the UI of the target site
  - operate search fields and other text input
- basic or digest authentication? cookies?
  - browser automatically sends credential
  - **session riding**
- POST requests?
  - manufacture a `<form>` instead of `<img>`

- study in late 2008: high-profile bank websites vulnerable
- DSL-Routers
  - disable firewall
  - reset wifi protection
  - enable UPnP
- browser-based port scanning
  - this is behind the corporate firewall

- disable cross-site POST requests
  - GET requests should by definition never change persistent state
  - there is a Firefox plugin for that
- never authenticate a change of persistent state by cookie only
- pass an additional credential
  - session ID in URL, edit tokens

## Log in

---

Don't have an account? [Create an account.](#)

You must have cookies enabled to log in to OSWiki.

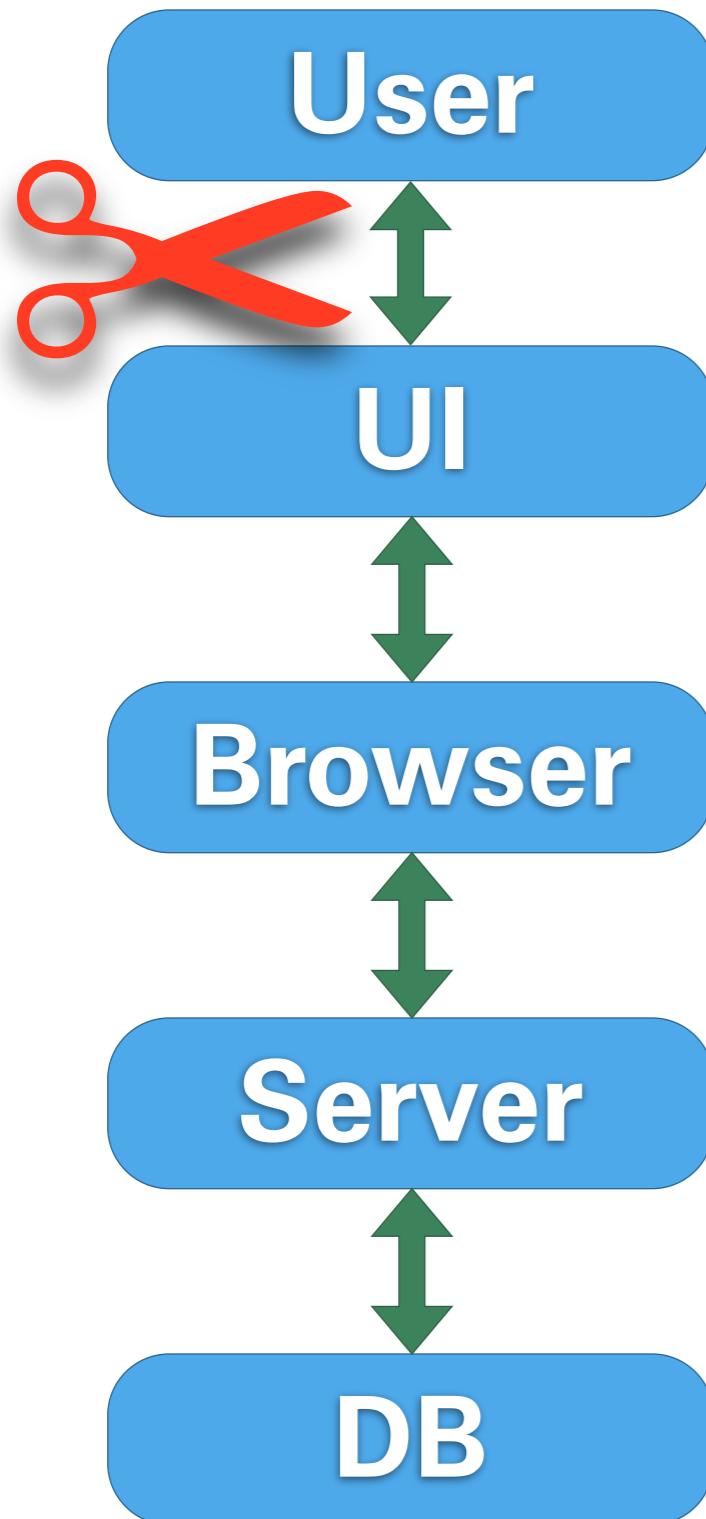
Username:

Password:

Remember my login on this computer

Log in

E-mail new password



- **goal:** mislead the user to not seeing what's actually happening
- nothing going on behind your back
- the internal state of the browser is properly displayed
- but you don't notice...



www . paypa<sup>1</sup> . com

CYRILLIC SMALL  
LETTER A (U+0430)

LATIN SMALL LETTER A  
(U+0061)

www . paypa<sup>1</sup> . com

**homograph attack**

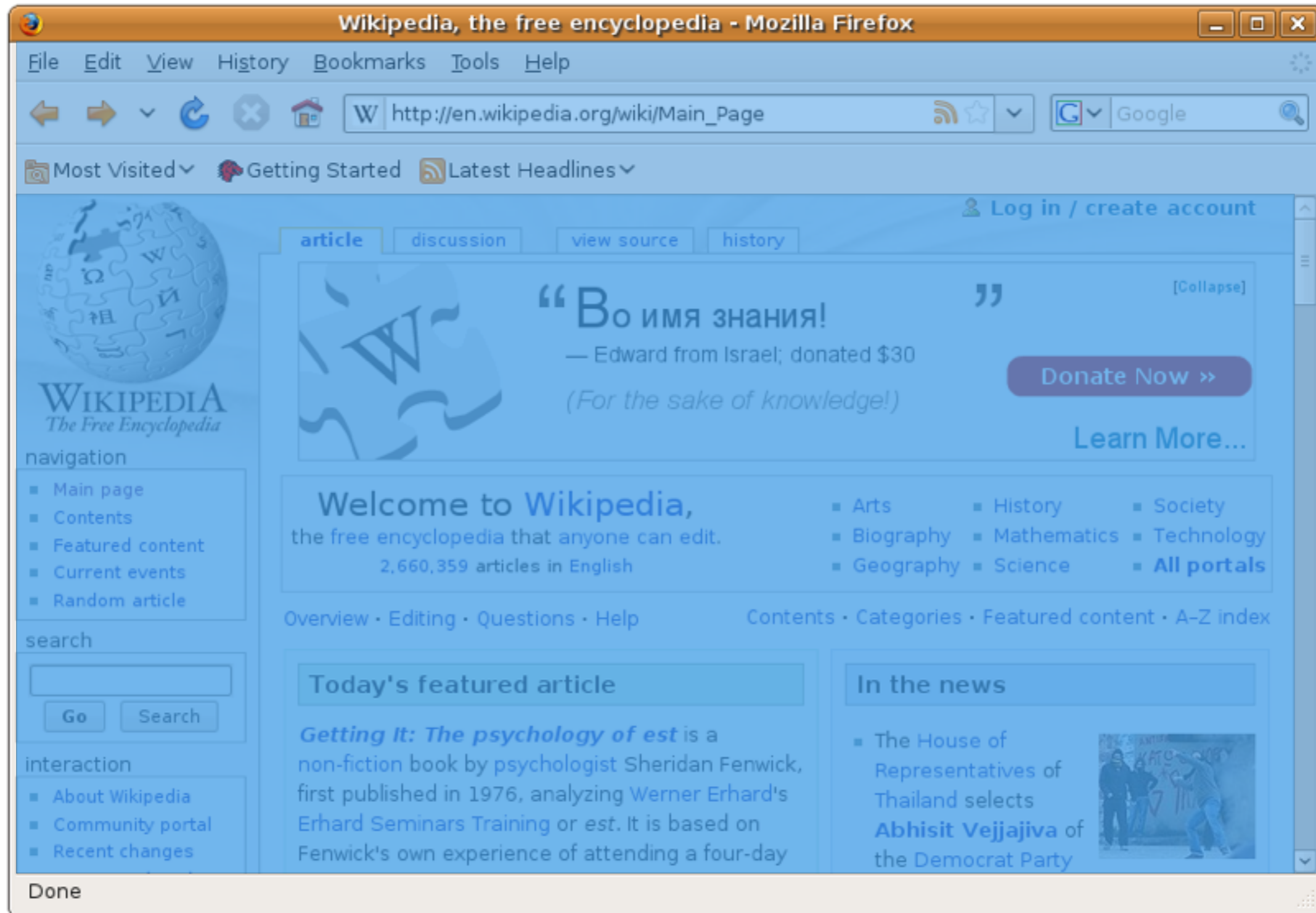
FRACTION SLASH  
(U+2044)

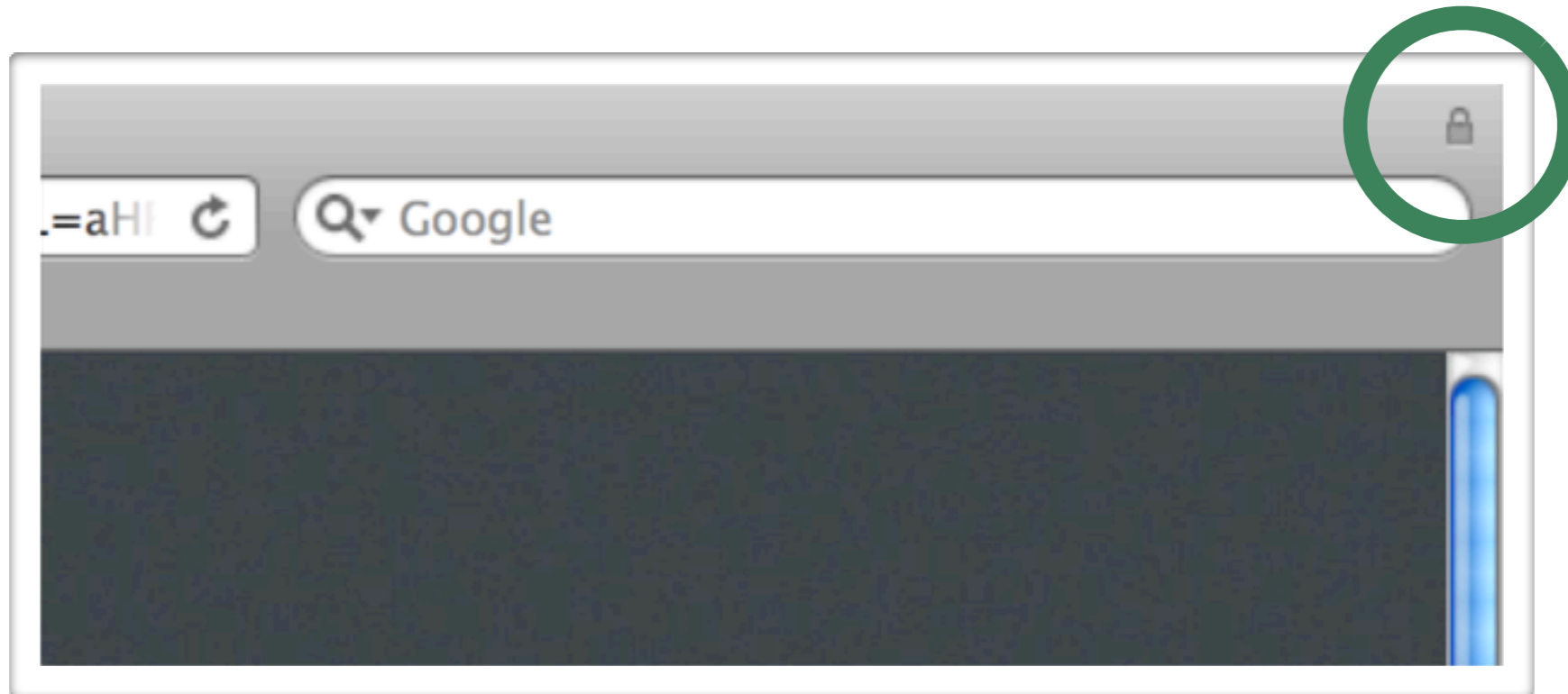
https://www.bank.com/account/login.ab.cd

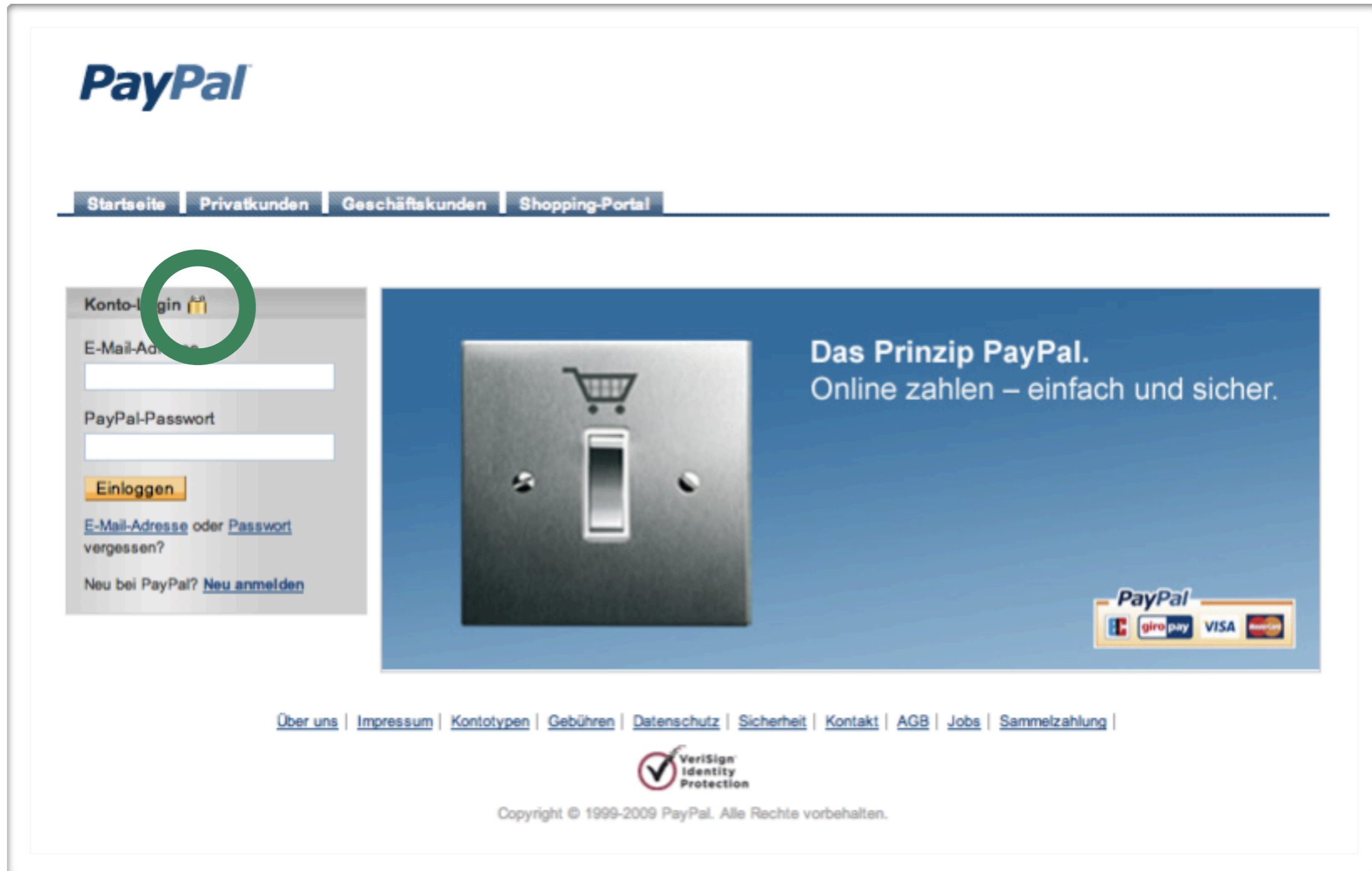
www.bank.xn--comaccountlogin-uh0iha.ab.cd

https://www.bank.com/account/login.ab.cd

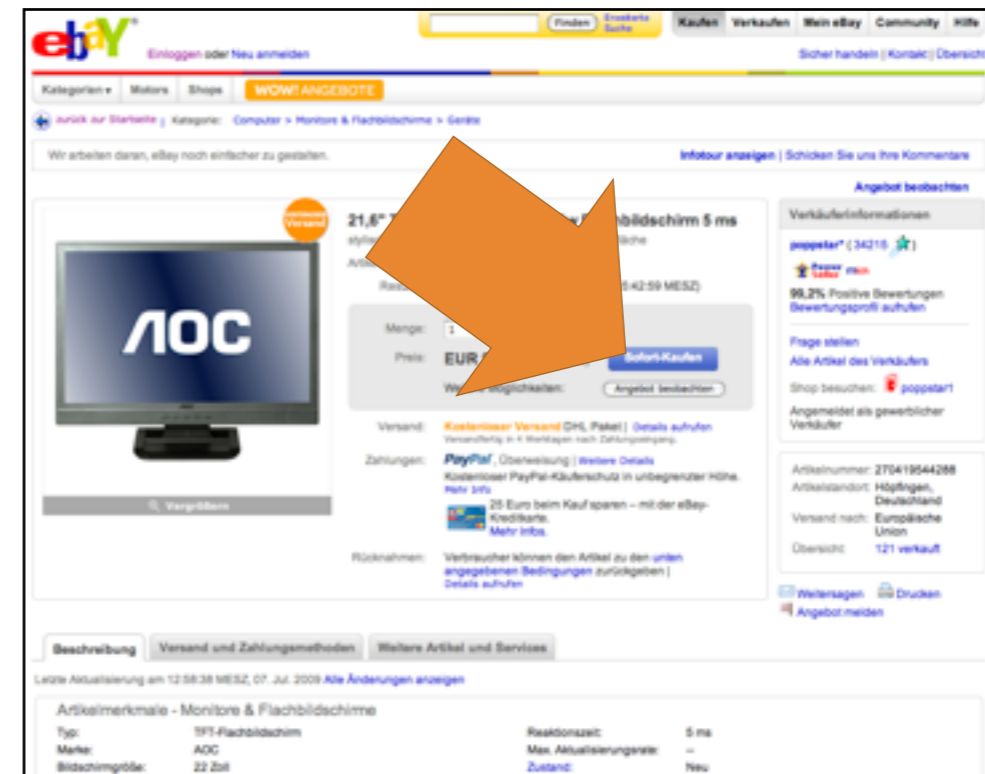
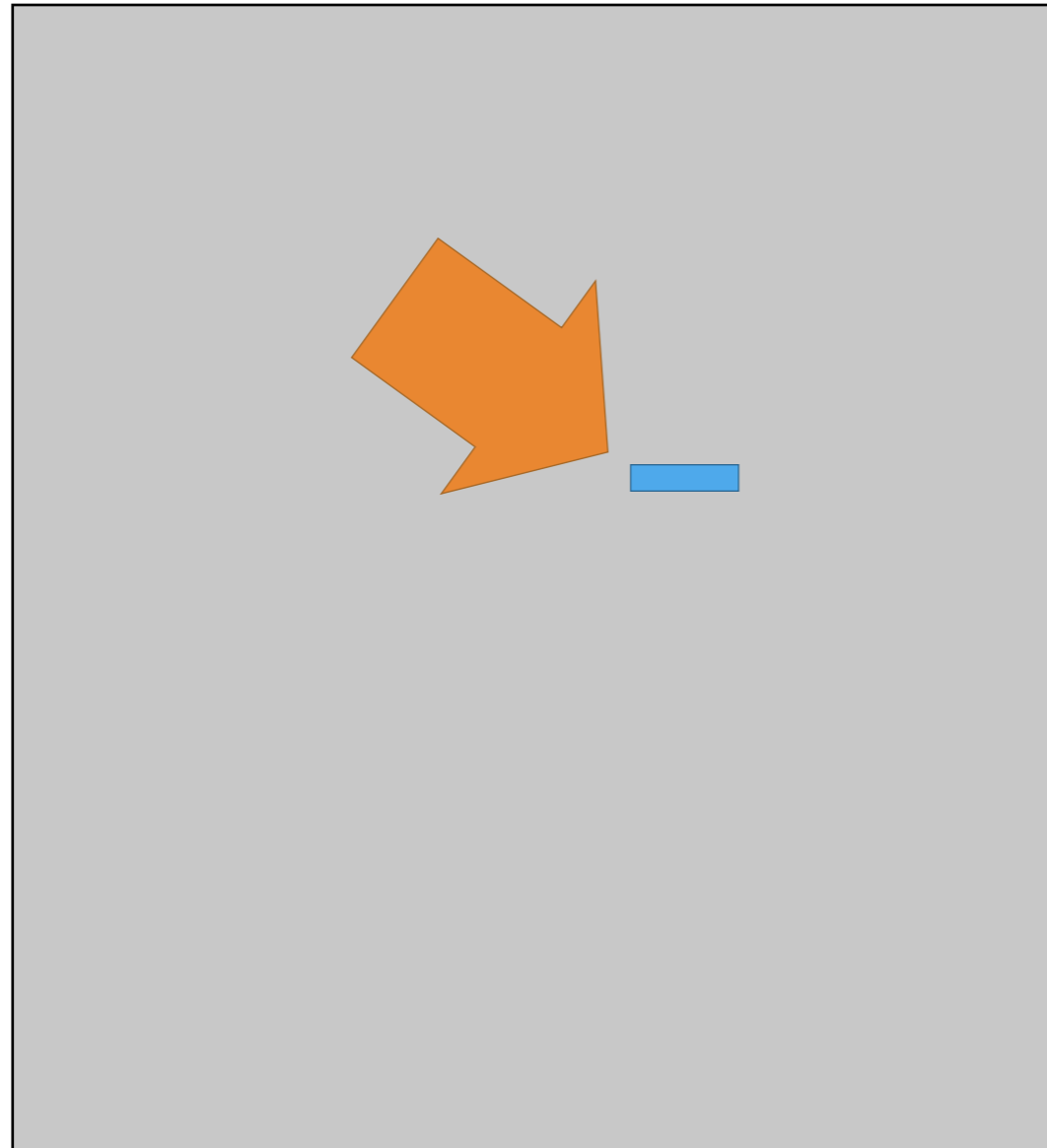
www.bank.com







The image shows a screenshot of the PayPal website. At the top left is the PayPal logo. Below it is a navigation bar with links for 'Startseite', 'Privatkunden', 'Geschäftskunden', and 'Shopping-Portal'. On the left side, there is a login form titled 'Konto-Login' with a green circle around the 'Konto-Login' text. The form includes fields for 'E-Mail-Adresse' and 'PayPal-Passwort', an 'Einloggen' button, and links for 'E-Mail-Adresse oder Passwort vergessen?' and 'Neu bei PayPal? Neu anmelden'. To the right of the login form is a large blue banner with the text 'Das Prinzip PayPal. Online zahlen – einfach und sicher.' and a shopping cart icon. Below the banner are logos for PayPal, giro pay, VISA, and MasterCard. At the bottom of the page, there is a footer with links for 'Über uns', 'Impressum', 'Kontotypen', 'Gebühren', 'Datenschutz', 'Sicherheit', 'Kontakt', 'AGB', 'Jobs', and 'Sammelzahlung'. Below the footer is the VeriSign Identity Protection logo and the copyright notice 'Copyright © 1999-2009 PayPal. Alle Rechte vorbehalten.'



- this only works when logged in
  - always log out explicitly
  - do not use persistent logins
- you may want to check whether your password manager autofills inside frames

Is everything lost?

**Yes**