- Above: iframe containing a hidden, **invisible** website with actionable content (which the user should click on)

- Under: a decoy website

-> Users only see the decoy website

PoC with **Burp Clickbandit**: Burp => Burp Clickbandit => Copy ClickBandit to clipboard => open developer console in the web page we want to test => paste, enter => enable Sandbox iframe (allow-scripts OR allow-forms) => Start => perform clicks to record the attack => Finish => Save (HTML is saved to clipboard)

## **Clickjacking with prefilled form input**

Websites might require text before form submission

* **Use get Request** (with parameter in URL) to fill the form before POST-Request

## **Combining clickjacking with a DOM**[**XSS**](https://portswigger.net/web-security/cross-site-scripting)**attack**

Fool the user clicks on the button or link that consequently executes the DOM XSS attack**.**

**Check for:**

* Verwendung von "frame-ancestors"-Direktive in Content Security Policy (CSP)

Content-Security-Policy: frame-ancestors 'none';

Content-Security-Policy: frame-ancestors 'self';

Content-Security-Policy: frame-ancestors normal-website.com;

* Verwendung von X-Frame-Options Header

X-Frame-Options: deny

X-Frame-Options: sameorigin

X-Frame-Options: allow-from https://normal-website.com