

Automated Web Attacks Using Headless Browsers

Dima Bekerman & Ben Herzberg

September 2016

ben.about()

```
> ben.history  
<· ["PT","Dev"]  
> ben.employer  
<· "Imperva Incapsula"  
> ben.positionX  
<· "Sec. Research Manager"  
> ben.social  
<· {"TWT": "@KernelXSS", "LNK": "Ben Herzberg"}
```



dima.about()

```
> dima.history  
<· ["Machine Learning", "Sec. Analysis"]  
> dima.employer  
<· "Imperva Incapsula"  
> dima.positionX  
<· "Security Researcher"  
> dima.social  
<· {"TWT": "@_unxmaster", "LNK": "Dima Bekerman"}
```



Remember...

What happens in Novi Sad stays in Novi Sad

A long, empty rectangular search bar with a thin gray border. On the right side of the bar is a small, colorful microphone icon, indicating voice search functionality.

Google Search

I'm Feeling Lucky

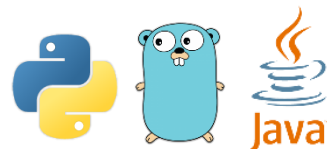
Headless browser

A headless browser is a web browser without a graphical user interface. It accesses web pages but doesn't show them to any human being, it used to provide the content of web pages to other programs.

https://en.wikipedia.org/wiki/Headless_browser

No JavaScript
No Dom
No Styling
No Layouts

HTTP library



Web Browser



Command line tool



Headless Browser



Web browser engine

A web browser engine is a program that renders marked up content and formatting information

Blink

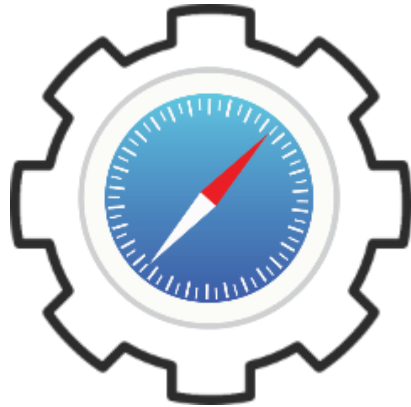


Chrome 28+

Opera 15+

Android 4.4+

WebKit



Safari

iOS

Chrome <27

Gecko



Firefox

IceWeasel

SeaMonkey

Trident



Internet Explorer

Windows Mobile

EdgeHTML

Presto



Opera

Opera Mini

Opera Mobile

What is DOM – Document Object Model?

Is the
HTML
you write
DOM?

Nope

The HTML you write is
parsed by the browser
and turned into the DOM

Is the
'View-
Source'
DOM?

Nope

View Source just shows
you the HTML that
makes up that page.
It's probably the exact
HTML that you wrote.

Is the
code in
DevTool
DOM?

Yep

DevTools panel
shows you a visual
representation of the
DOM.

Source Code

```
<div id="container"></div>

<script>
    var container = document.getElementById("container");
    container.innerHTML = "Hello World";
</script>
```

DOM representation

```
<div id="container">Hello World</div>
```

Page processing

- JavaScript
- CSS
- SVG
- MathML
- WebGL



Modernizer web features test

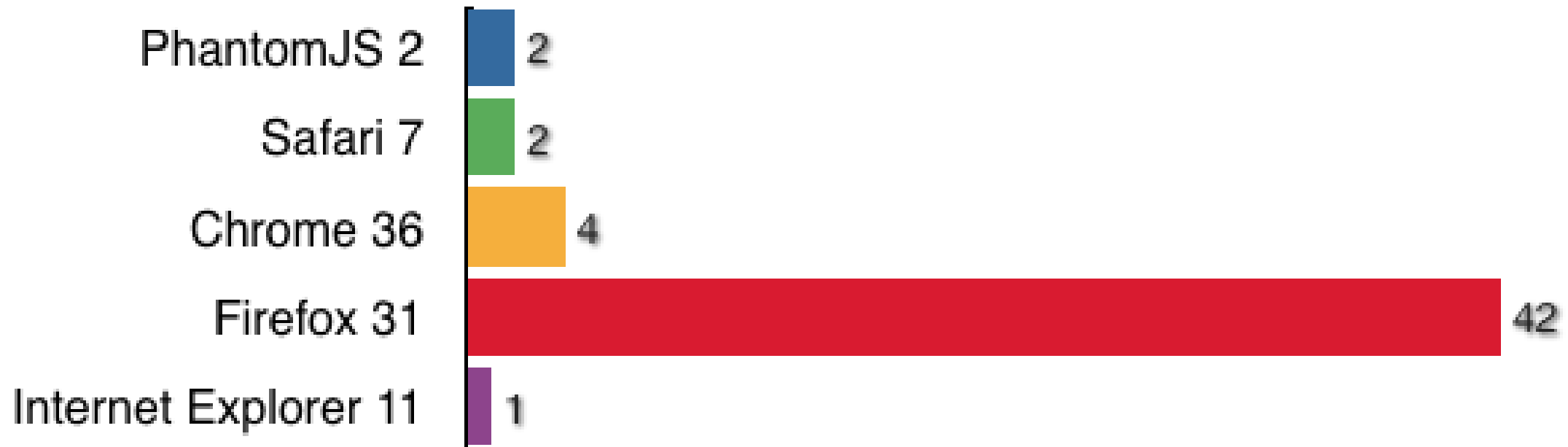


<pre>input{} ◦ autocomplete: true ◦ autofocus: true ◦ list: true ◦ placeholder: true ◦ max: true ◦ min: true ◦ multiple: true ◦ pattern: true ◦ required: true ◦ step: true inputtype{} ◦ search: true ◦ text: true ◦ url: true ◦ email: true ◦ date: true ◦ datetime: true ◦ month: true ◦ week: true ◦ time: true ◦ number: true ◦ range: true ◦ color: true htmlimports: false adownload: true ambientlight: false applicationcache: true audio: false audioloop: false audiopreload: false webaudio: false batteryapi: false battery-api: false lowbattery: false blobconstructor: true blob-constructor: true canvas: true canvasblending: true todataurljpeg: true todataurlpng: true todataurlwebp: false canvaswinding: true</pre>	<pre>borderradius: true boxshadow: true boxsizing: true csscalc: true checked: true csschunit: true csscolumns{} ◦ width: true csscubic: true cssfontface: true cssgradient: true csshsla: true cssinert: true cssinvalid: true lastchild: true cssmask: true mediaqueries: true multiplebgs: true nthchild: true objectfit: false</pre>	<pre>textshadow: true csstransforms: true csstransforms3d: true preserve3d: false userselect: true cssvalid: true cssvhunit: true cssvmaxunit: true cssvminunit: true cssvwunit: true willchange: true wrapflow: false customprotocolhandler: false customevent: false dart: false dataview: true classList: true createElement: true createElementNS: true dataset: true documentfragment: true hidden: true microdata: false mutationobserver: true passiveeventlistener: true bdi: true datalistelement: true details: true outputelement: true picture: false progressbar: true meter: true ruby: true template: false texttrackapi: false track: false unknownelements: true emoji: true es5array: true es5date: true es5function: true es5object: true strictmode: true es5string: true</pre>	<pre>capture: false fileinput: true fileinputdirectory: false formattribute: true formvalidationapi: true formvalidationmessage: false formvalidation: false localizednumber: false requestautocomplete: false fullscreen: true gamepads: false geolocation: false hashchange: true hiddenscroll: false history: true ie8compat: false</pre>	<pre>localStorage: true sessionStorage: true webSQLDatabase: true stylescoped: false svg: true svgasimg: true svgclippaths: true svgfilters: true svgforeignobject: true inlinesvg: true smil: true templatestrings: false textareamaxlength: true touchevents: true typedarrays: true unicode: true unicoderaange: true bloburls: true urlparser: false urlsearchparams: false userdata: false vibrate: false video: false videoautoplay: false videocrossorigin: false videoloop: false videopreload: false vml: false webintents: false webanimations: false webgl: false webglextensions: false peerconnection: false datachannel: false getusermedia: false websockets: true websocketsbinary: true framed: false sharedworkers: true webworkers: true transferables: true xdomainrequest: false csshyphens: false softhyphens: true</pre>
--	--	---	--	---

svg: true
svgasimg: true
svgclippaths: true
svgfilters: true
svgforeignobject: true

ECMAScript test262

Failures in test262





PhantomJS

- Based on WebKit
- Cross Platform
- Portable
- Scriptable with JavaScript
- Good for WebUI automation



SlimersJS

Based on Gecko



TrifleJS

Based on Trident



Awesomium

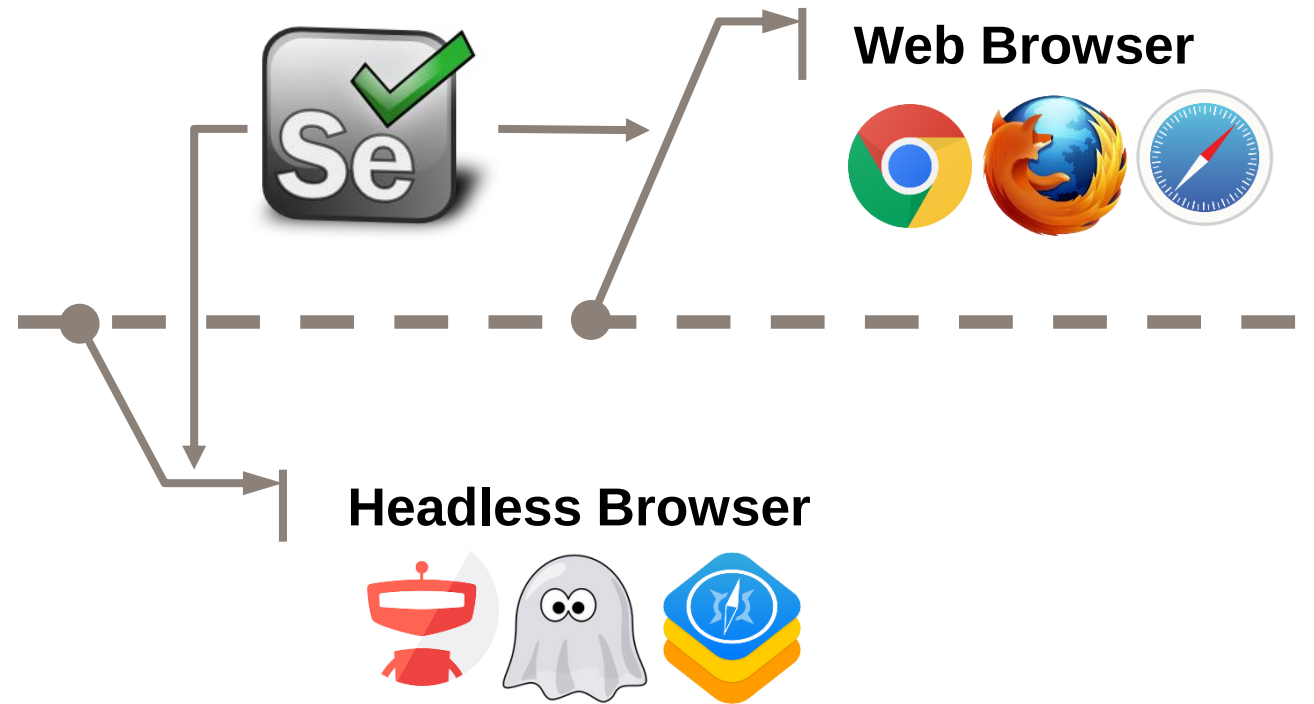
- Chromium based
- V8 – JavaScript Engine
- Programmable with .NET, C++
- Compiled to cross platform app
- Can be used as a module



Chromium Embedded Framework



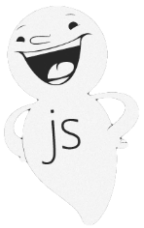
WebKit .NET





Selenium / WebDriver

- Selenium is a browser control and automation system
- Manipulating a real web browser using a browser driver
- Manipulated browser must be installed
- Host OS should have a GUI framework

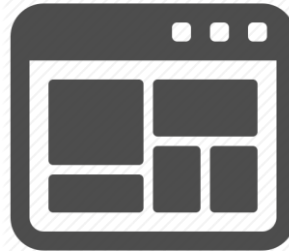


The world of two contexts



Browser Context

- Language depends on framework
- Compiled / evaluated inside framework interpreter
- Has access to OS



Page Context

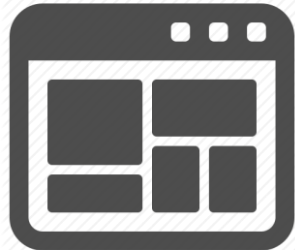
- JavaScript
- Evaluated by the web engine
- Has access to DOM objects

The world of two contexts



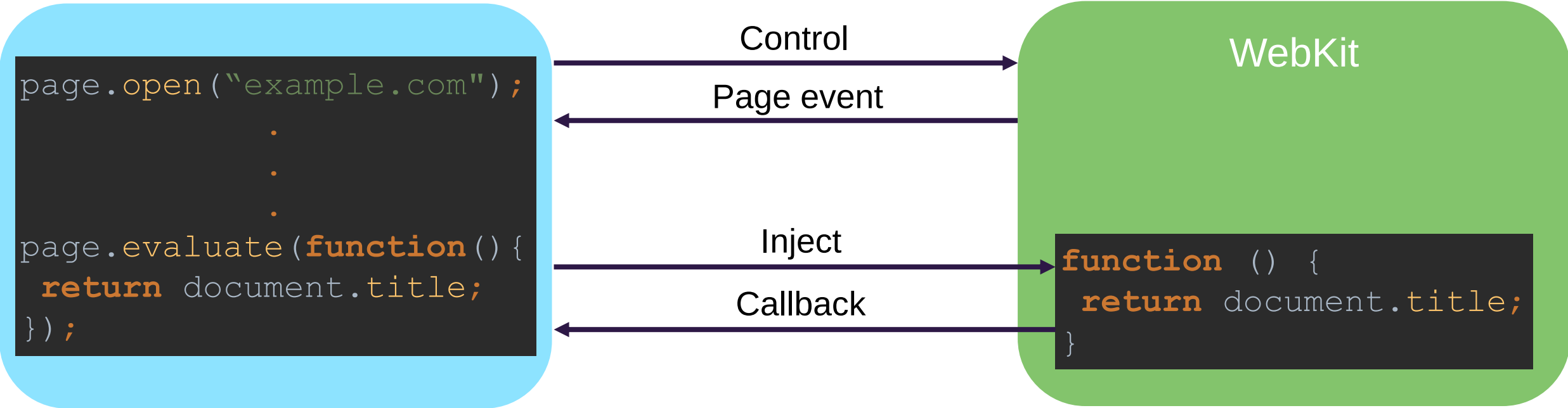
Browser Context

my_script.js



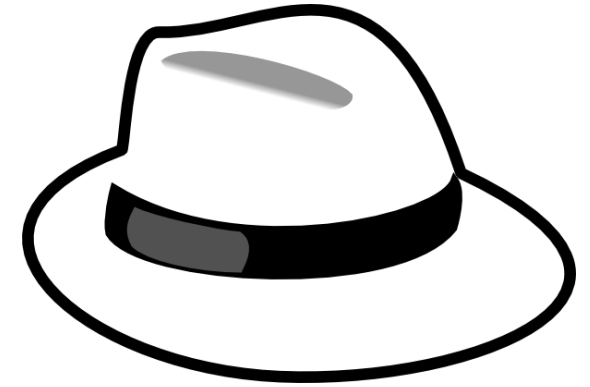
Page Context

example.com



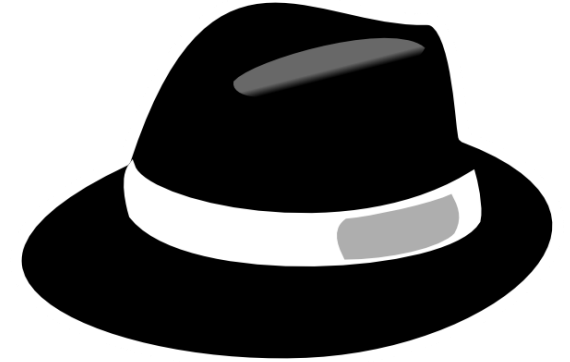
Use Cases

- Test automation
- Crawl websites
- Render dynamic web page to static HTML
- Inject and execute arbitrary JavaScript code
- Perform actions
- Taking screen shots
- Monitor performance



Use Cases

- Content scraping
- Click fraud
- Fuzzing
- Bidding abuse
- Login brute force
- Botnet
- Sophisticated DDoS attacks



Capture screen to image

Source Code

```
var page = require('webpage').create();

page.open('http://www.google.com', function (status) {
    page.render('google.png');
    phantom.exit();
});
```



Grabbing text

Source Code

```
//Page title  
console.log(page.title);  
  
// View-source  
console.log(page.content);  
  
//Content without HTML tags  
console.log(page.plainText);
```

Handle console and alerts

Source Code

```
//Handle console
page.onConsoleMessage = function(msg) {
    console.log('CONSOLE: ' + msg);
};

//Handle alert
page.onAlert = function(msg) {
    console.log('ALERT: ' + msg);
};
```

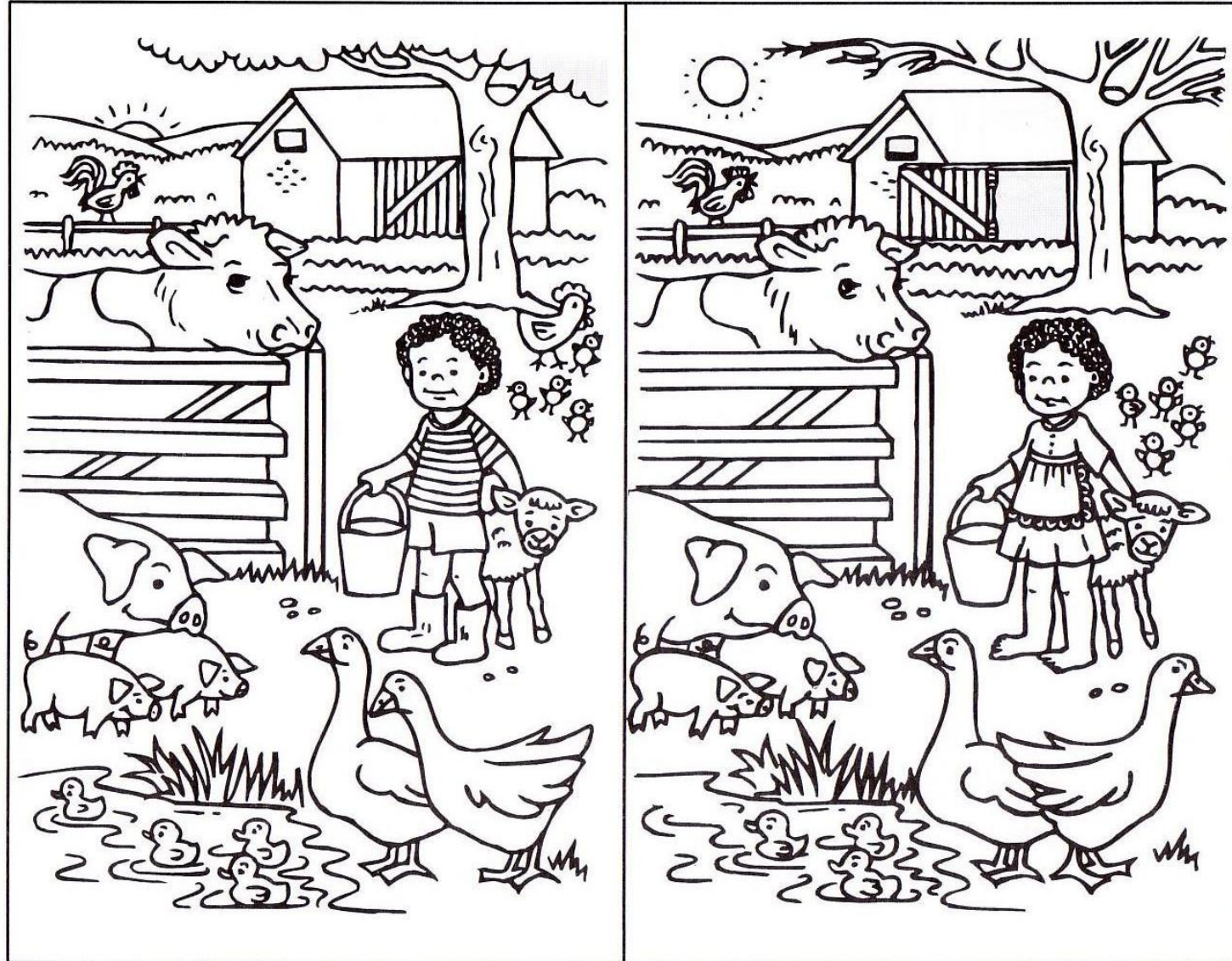
A long, empty rectangular search bar with a thin grey border. On the right side of the bar is a small, colorful microphone icon, indicating voice search functionality.

Google Search

I'm Feeling Lucky



Spot the differences





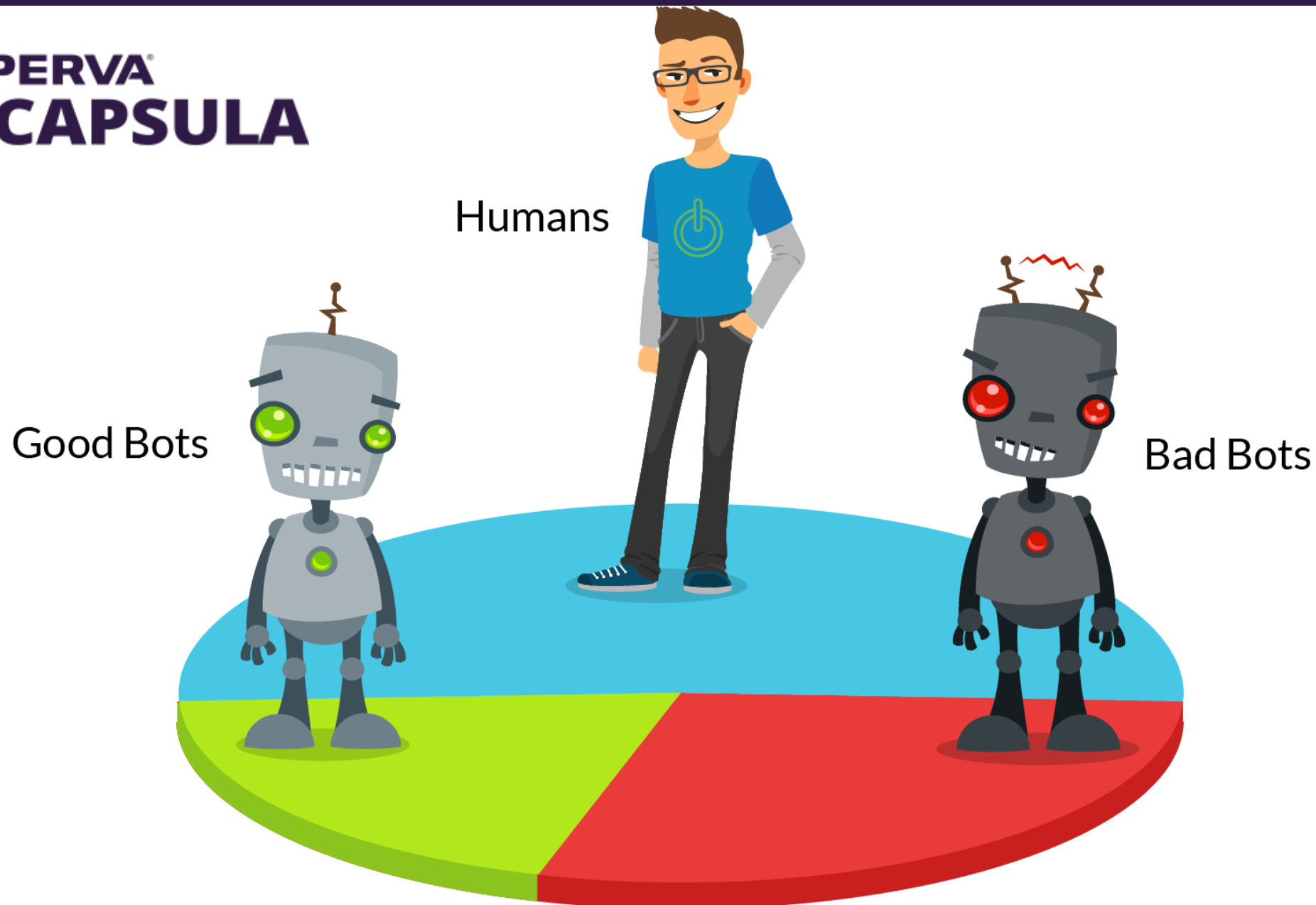
Awesomium

Android WebView



Source Code

```
webView.setWebViewClient(new WebViewClient() {  
    @Override  
    public void onPageFinished(WebView view, String url) {  
        browser.loadUrl(  
            "javascript:window.HTMLOUT.processHTML(  
                document.getElementsByTagName('html'));" );  
    }  
});
```



Scenarios from the wild

- SEO tools scrape for dynamic content
- E-commerce abuse
- DDoS attacks from a mobile botnet

Хвала вам!

THANK YOU!



@KernelXSS



@_unxmaster



Ben Herzberg



Dima Bekerman



hben@incapsula.com



dima@incapsula.com

IMPERVA®