



IMT Atlantique

Bretagne-Pays de la Loire
École Mines-Télécom

KEEP YOUR DATA UNDER CONTROL

PAUL BESLIN
JOANNE DUMONT
OUSSAMA EL MOATAMID
XI SONG

Ce powerpoint n'a jamais servi à présenter le projet: il sert d'extension à celui qui nous a servi pour le forum et rentre dans les détails des sujets les plus subtils sur lesquels nous sommes penchés.

1. CONTEXTE ET OBJECTIFS

2. PROJET HÉRITÉ

3. TRAVAIL DE L'ANNÉE - DÉTAILS

1. Réflexion sur les moteurs de recherche
2. Access lists et sécurité
3. Fonctionnement de l'extension

4. CONCLUSION ET OUVERTURES



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

CONTEXTE ET OBJECTIFS



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

LA QUESTION DE LA SÉCURITÉ DES DONNÉES

Chaque jour :



700 millions+ de
tweets



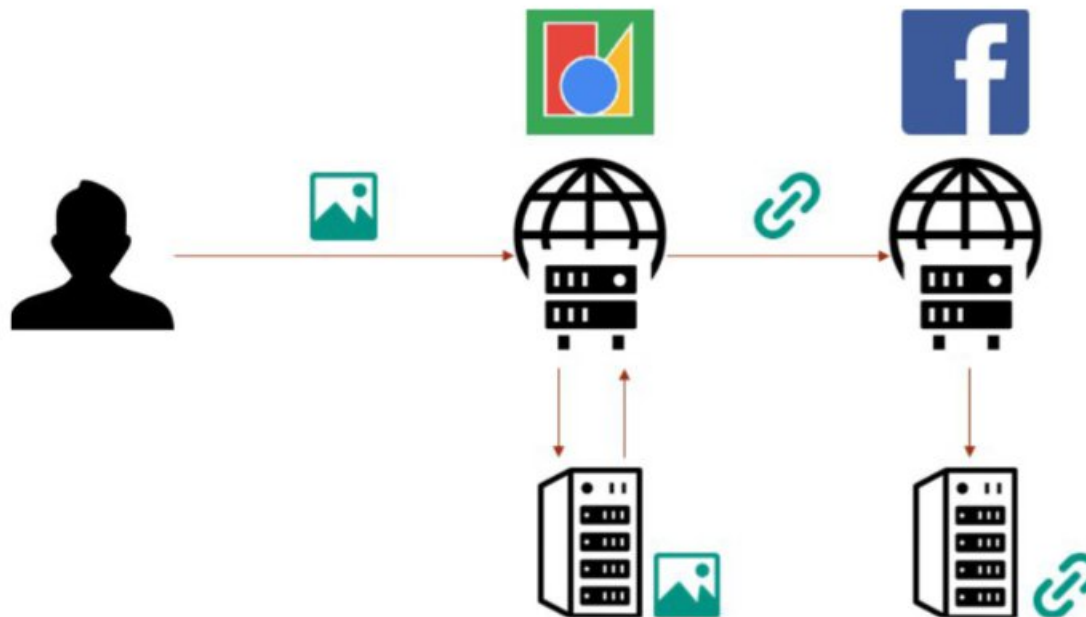
100 millions+
de photos et
vidéos



2 millions de
publications

LE PROJET

Une extension de navigateur qui permettrait de remplacer ses données par des liens d'accès en lecture.



UNE PROJET SUR PLUSIEURS ANNÉES



bastantoine / [KeepYourDataUnderControl](#)

Public archive

- c'est la deuxième année du projet
- les deux élèves de l'année précédente ont réfléchi aux concepts à appliquer pour mettre en place le projet: nous avons pu démarrer sur des fondations déjà posées (lien sur le README principal du github)
- il restait cependant beaucoup à faire (et c'est toujours le cas après notre passage !)



MT Atlantique
Bretagne - Pays de la Loire
Finistère - Morbihan - Vendée

TRAVAIL DE L'ANNÉE - DÉTAILS



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

Les slides qui suivent ont pour but d'étendre les informations fournies dans le github et le powerpoint de présentation du forum.

RÉFLEXIONS SUR LES MOTEURS DE RECHERCHE



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

BROWSER ARCHITECTURE OVERVIEW

Browser Process



Renderer Process



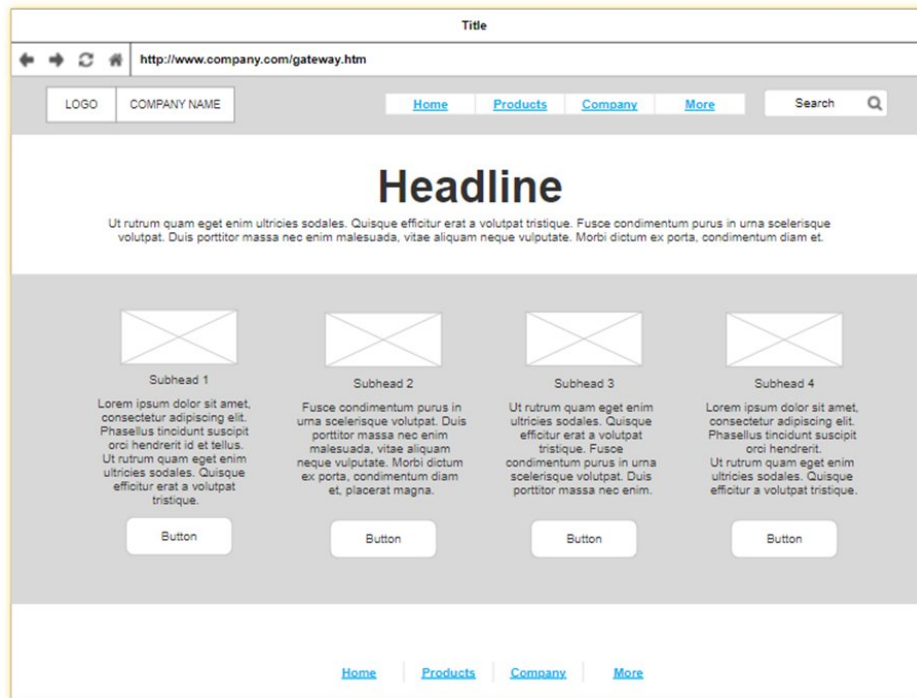
Plugin Process



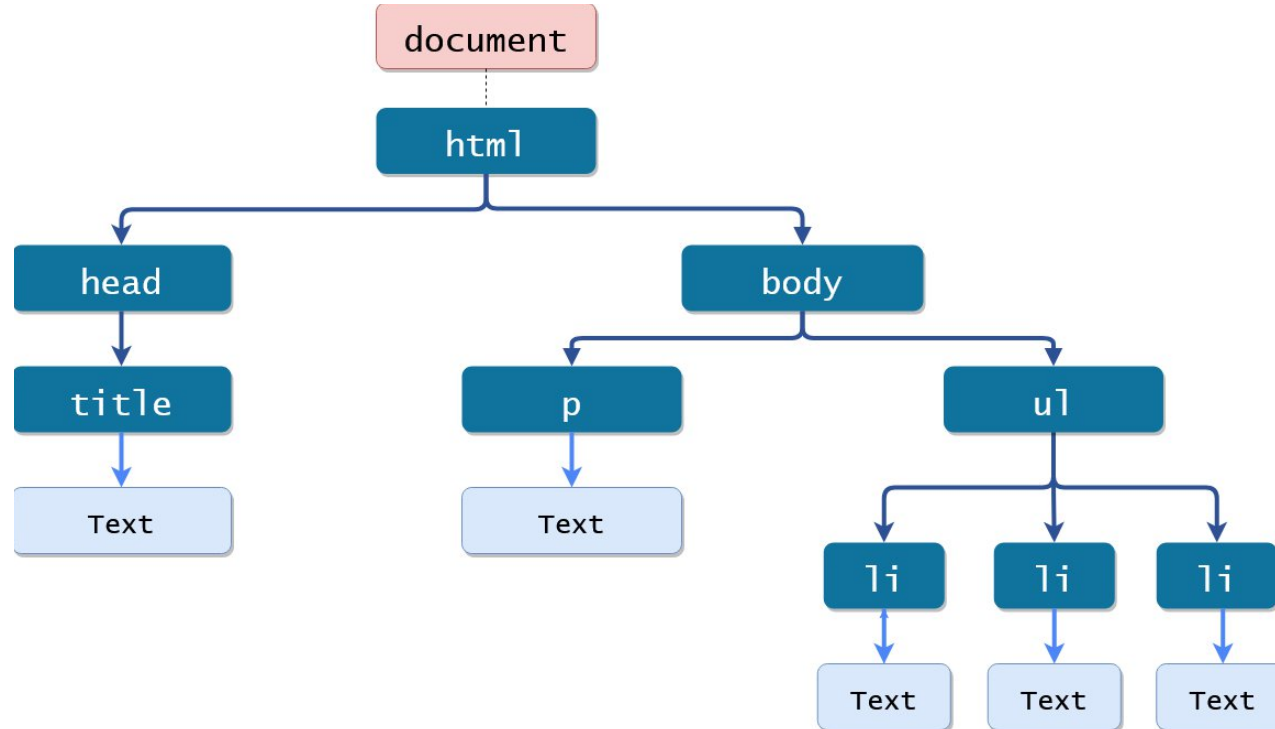
Extension Process



GPU Process

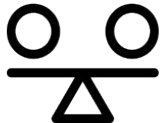


SHARED DOM



EXTENSION PROCESS ISOLATION

Stability



Performance



Security



SCRIPT INJECTION

Programmatically

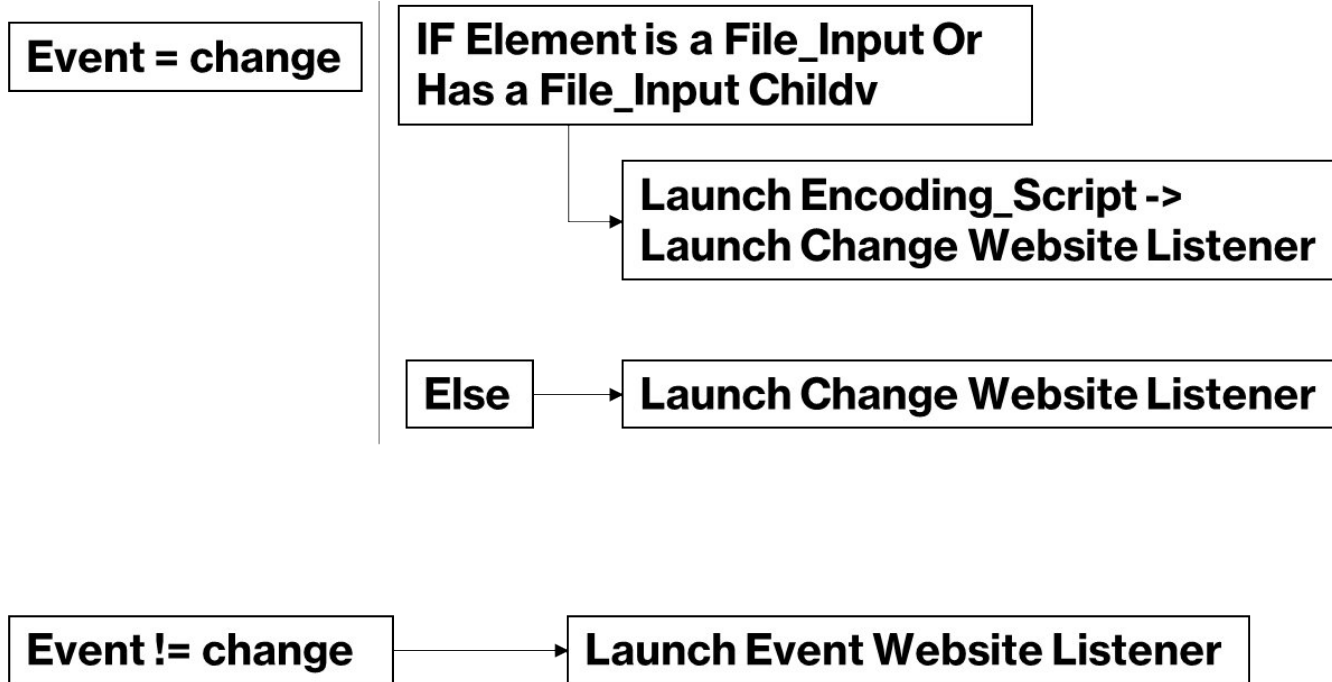
```
{  
  "name": "My extension",  
  ...  
  "permissions": [  
    "activeTab"  
  ],  
  ...  
}
```

```
chrome.runtime.onMessage.addListener(  
  function(message, callback) {  
    if (message == "runContentScript"){  
      chrome.tabs.executeScript({  
        file: 'contentScript.js'  
      });  
    }  
  }  
);
```

Declaratively

```
{  
  "name": "My extension",  
  ...  
  "content_scripts": [  
    {  
      "matches": ["http://*.nytimes.com/*"],  
      "css": ["myStyles.css"],  
      "js": ["contentScript.js"]  
    }  
  ],  
  ...  
}
```

IMAGE ENCODING SOLUTION



ACCESS LISTS ET SÉCURITÉ



IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

SÉCURISER LES ACCÈS À LA BASE DE DONNÉES

Les access lists (ACL) dans ce projet, c'est:

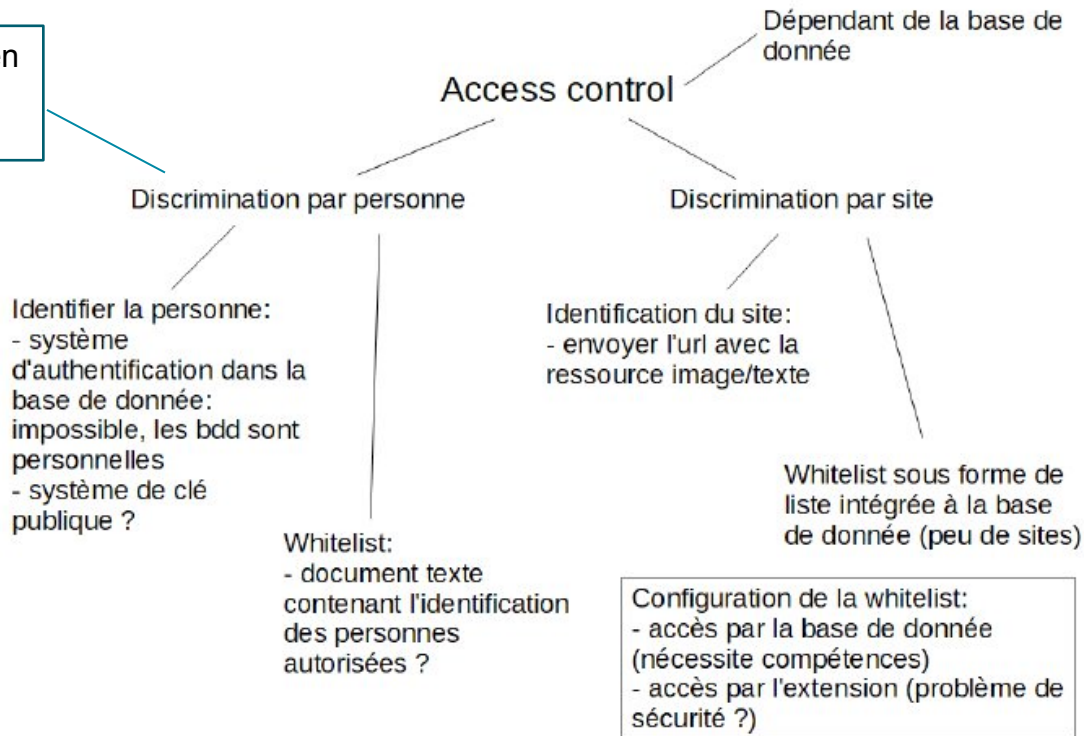
- une envie d'approfondir la dimension de sécurité des données: on les a dans une seule base qui est à nous mais en ligne, et maintenant comment fait-on pour restreindre qui y a accès ?
- une introduction à la question de sécurité des bases de données

SÉCURISER LES ACCÈS À LA BASE DE DONNÉES

Réflexion sur la stratégie à adopter

Les access lists, c'est la white list et la black list: les listes des personnes/entités qui ont l'autorisation ou l'interdiction d'accéder à une ressource.

Inutile, les réseaux s'en servent déjà (listes d'amis, etc)



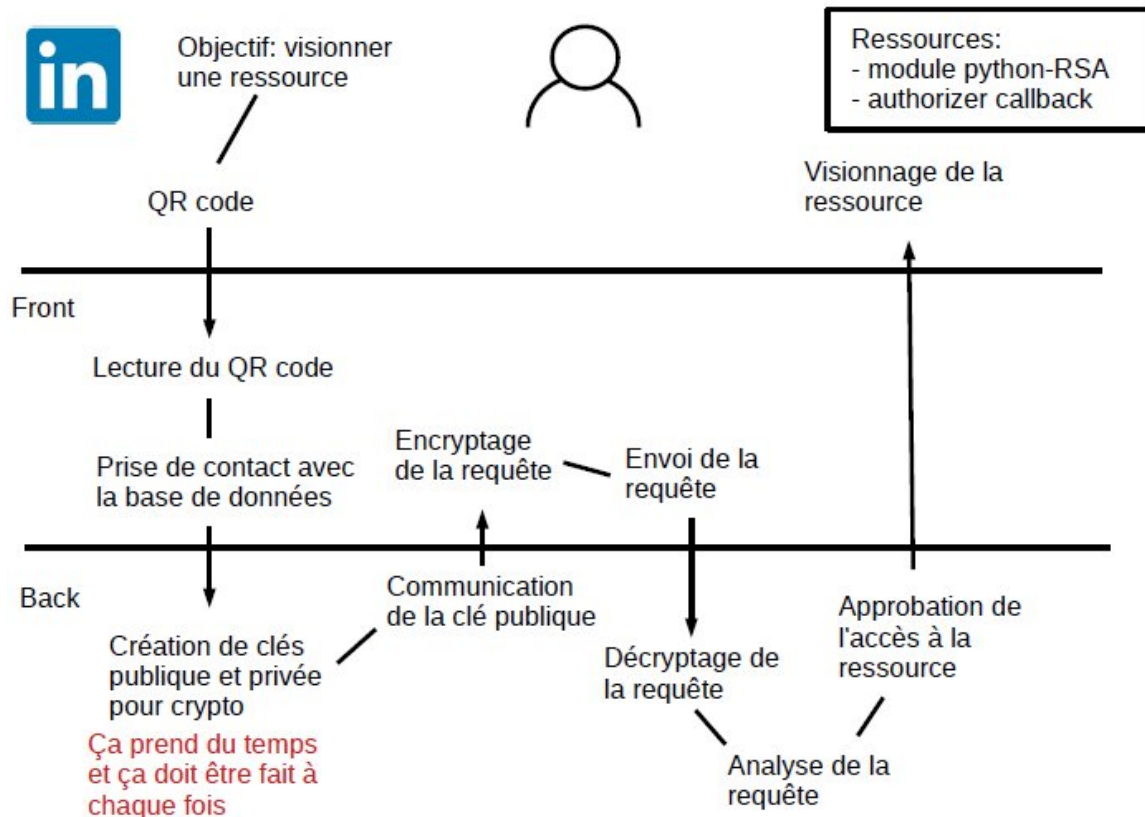
DISCRIMINATION PAR SITE

Première idée trop complexe

Problèmes:

- complexe
- long à effectuer à chaque fois qu'il y a communication entre l'extension et la bdd
- ça ne résout pas le problème: ce qu'on fait réellement ici, c'est s'assurer qu'on communique bien avec l'extension: c'est bien, mais ce n'est pas ce qu'on veut et un peu overkill

Acquis grâce à cette fausse-route: quelques notions de crypto (système RSA).



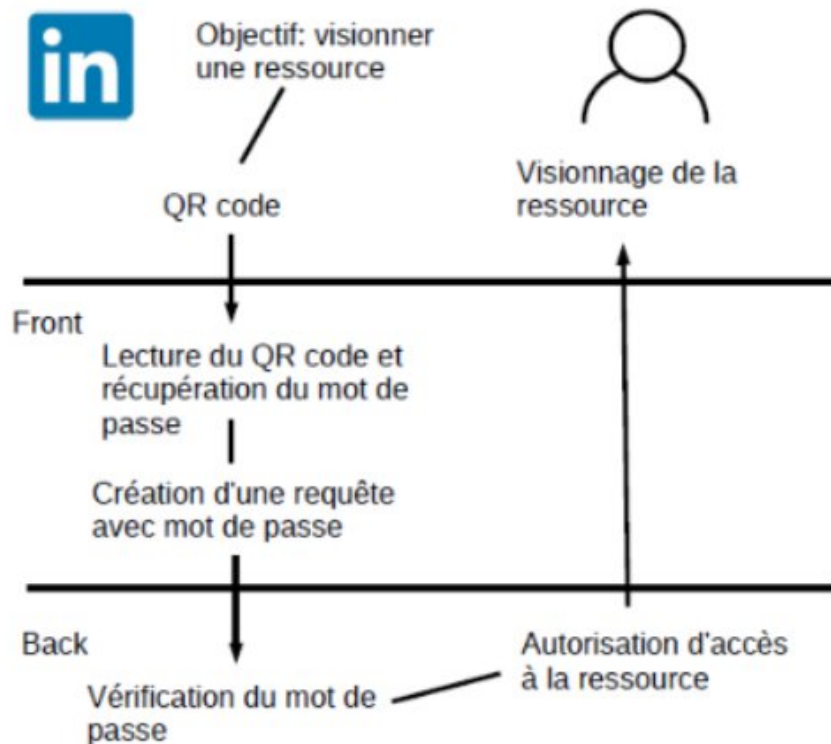
DISCRIMINATION PAR SITE

Idée retenue

Nouvelle idée, plus simple et en deux parties:

- ajouter une barrière à l'accès à la ressource, sous forme d'un mot de passe par exemple: le but est qu'il n'y ait pas d'autre moyen pour avoir accès à la ressource que d'avoir l'url déposé sur le réseau

- créer une whitelist de réseaux: on ne peut avoir accès à la ressource que si on tente de l'obtenir depuis le site sur lequel son url a été déposé, ou un autre explicitement autorisé par le propriétaire



LE RÉSULTAT

Identifiants des ressources

Dans les bases de données "exemple" de petits projets, les identifiants des ressources sont souvent des entiers, dans l'ordre. C'est une faille de sécurité: avec une url, on trouve les autres.

id

GET <http://mabdd.com/ressource/23>



GET <http://mabdd.com/ressource/24>

GET <http://mabdd.com/ressource/25>

...

uuid

GET <http://mabdd.com/ressource/5612-4897-3749-abx>



?

Pas besoin de mot de passe finalement: un meilleur identifiant fait l'affaire.

LE RÉSULTAT

Identifier le site d'origine

GET <http://mabdd.com/ressource/5612-4897-3749-abx>



POST <http://mabdd.com/ressource/5612-4897-3749-abx> {site d'origine}

Whitelist



~~*Whitelist*~~

Sorry, you have no permission to access this image.

Dans la requête
envoyée par
l'extension, le site
est précisé.

FONCTIONNEMENT DE L'EXTENSION

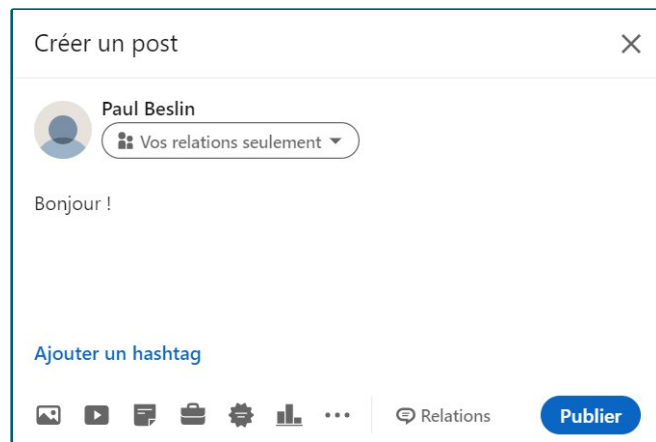


IMT Atlantique
Bretagne-Pays de la Loire
École Mines-Télécom

ORDRE DES OPÉRATIONS

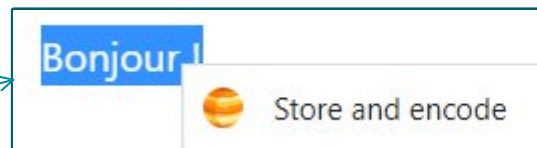
Texte - côté utilisateur

1) L'utilisateur entre son texte dans le formulaire de création de post.



The screenshot shows a 'Créer un post' (Create a post) dialog box. At the top, it says 'Créer un post' with a close button. Below that, the user's profile is shown as 'Paul Beslin' with a dropdown menu set to 'Vos relations seulement'. The text input field contains 'Bonjour !'. There is a link 'Ajouter un hashtag' and a row of icons for adding media (photo, video, document, etc.). At the bottom right is a blue 'Publier' button.

2) Il le sélectionne, effectue un clic droit et choisit l'option proposée par l'extension.



Intervention de l'extension

```
[KCoyD]http://localhost:5001/query/resource/28baa969-4bdb-4213-8519-3757c9d6c7f6[/KCoyD]
```

3) Le texte est encodé et l'utilisateur peut publier son post.

ORDRE DES OPÉRATIONS


Texte - côté extension

background.js

```
//https://stackoverflow.com/a/61038472
chrome.contextMenus.create({
  title: "Store and encode",
  contexts: ["selection", "image"],
  id: "encode"
});

chrome.contextMenus.onClicked.addListener(sendEncodingRequest);
```

A l'ouverture du navigateur, une action de menu contextuelle est créée et reliée à un listener.

Emoji	Windows+Point
Couper	Ctrl+X
Copier	Ctrl+C
Coller	Ctrl+V
Coller en texte brut	Ctrl+Maj+V
Tout sélectionner	Ctrl+A
Rechercher "e" avec Google	
Imprimer...	Ctrl+P
Correcteur orthographique	▶
Sens de l'écriture	▶
 Store and encode	
Inspector	

ORDRE DES OPÉRATIONS

Texte - côté extension

background.js

```
//If more context menus need to be created, another JSON key/value pair
//could be used to provide information on which action is expected.
function sendEncodingRequest(info, tab) {
    let dataType = undefined;
    let data = undefined;

    //Process a text.
    if (info.selectionText) {
        dataType = "text"; //enum?
        data = info.selectionText;
    }

    //Send captured data.
    chrome.tabs.sendMessage(tab.id, {
        type: dataType,
        selection: data
    });
}
```

Le listener réagit au clic sur "Store and encode" et envoie les détails (texte sélectionné et conteneur) pour qu'ils soient traités.

content-script2.js

```
switch (type) {
    case 'text':
        // We convert our text as a file
        blob = new Blob([data], { type: "text/plain;charset=utf-8" });
        form.append("file", blob, "tmp.txt");
        break;

    form.append("site", location.host);

    let settings = {
        // This url need to be changed to your own self storage
        "url": "http://localhost:5001/",
        "method": "POST",
        "crossOrigin": true,
        "timeout": 0,
        "processData": false,
        "mimeType": "multipart/form-data",
        "contentType": false,
        "data": form,
        "async": false
    };
    responseURL = await $.ajax(settings);
    return JSON.parse(responseURL).url;
}
```

Le texte est converti en fichier txt et envoyé en base de données. La BDD renvoie l'URL d'accès au texte.

ORDRE DES OPÉRATIONS

Texte - côté extension

text_encoding.js

```
function encode_text_url(url) {  
    return "[KCoyD]" + url + "[/KCoyD]";  
}
```

L'URL est encadrée de pseudo-tags qui faciliteront le décodage.

content-script2.js

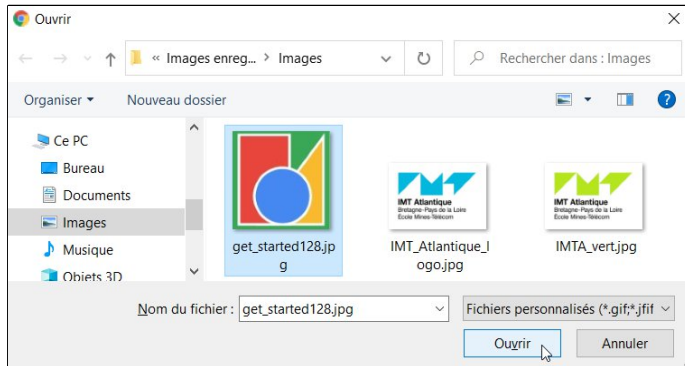
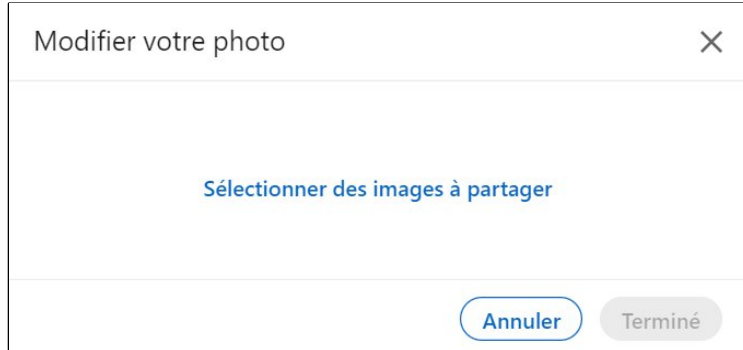
```
function replaceInPage(type, encodedData, $container) {  
    switch (type) {  
        case 'text':  
            $container.text(encodedData);  
            break;  
    }  
}
```

Le texte dans le conteneur est remplacé par l'URL encodée.

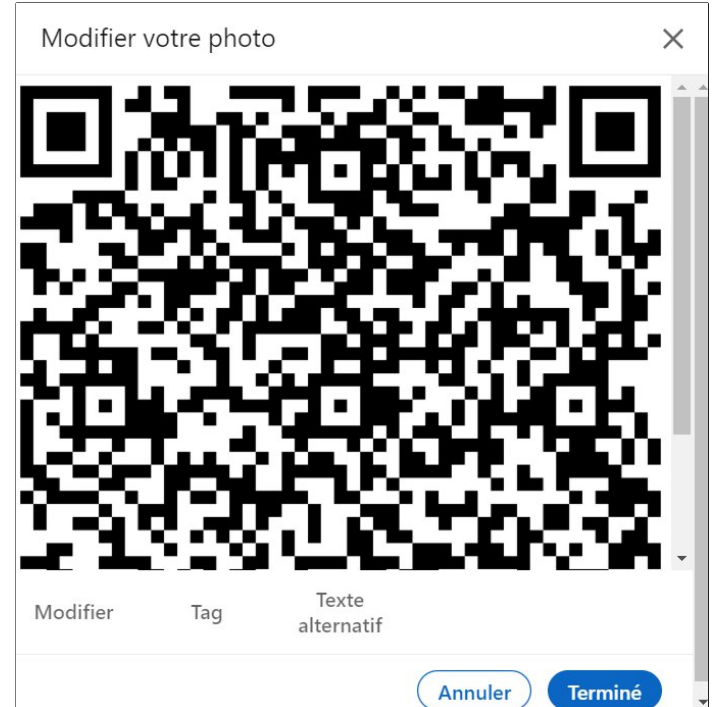
ORDRE DES OPÉRATIONS

Image - côté utilisateur

1) L'utilisateur sélectionne son image dans le formulaire de création de post.



2) L'image est encodée en QRCode et l'utilisateur peut publier son post.



ORDRE DES OPÉRATIONS

Image - côté extension

injected_script1.js

```
Element.prototype.addEventListener = function (a, b, c) {  
    if (c == undefined) c = false;  
  
    if (!this.eventListenerList) this.eventListenerList = {};  
    if (!this.eventListenerList[a]) this.eventListenerList[a] = [];  
  
    if (a == "change") {  
        let eventListener = e => {  
            if ((this.nodeName.toLowerCase() == 'input' && this.type.toLowerCase() == "file") || (this.querySelector('input[type=fi  
                window.postMessage({  
                    type: "Encode_Image",  
                    file: e.target.files[0]  
                }, "**");  
                setTimeout(() => {  
                    this.eventListenerList.change[1].listener(e);  
                }, 1000);  
            } else {  
                this.eventListenerList.change[1].listener(e);  
            }  
        }  
        this.eventListenerList[a].push(  

```

A l'ouverture du navigateur, un handler spécifique est injecté pour l'événement `input.onChange`, avant les handlers de la page. Cela permet de le rendre prioritaire.

ORDRE DES OPÉRATIONS

Image - côté extension

QrCode.js

```
encode(){  
  //TODO replace the following encoding algorithm by using a js library  
  if (this.link !== undefined) {  
  
    const qrcode = new Encoder();  
  
    //qrcode.write(this.link)  
    qrcode.write(this.link);  
    qrcode.setErrorCorrectionLevel(ErrorCorrectionLevel.H);  
  
    qrcode.make();  
  
    const base64Data = qrcode.toDataURL(15,12);  
    this.image = b64toBlob(base64Data.split(",")[1], 'image/png');  
  }  
}
```

Le handler injecté précédemment réagit à l'envoi d'une image et l'envoie en BDD.

Le lien récupéré ensuite (comme pour les textes) est transformé en QRCode.



MT Atlantique
Bretagne - Pays de la Loire
Enseignement Supérieur

content-script.js

```
//Image Replacement  
const finalFile = new File([data], 'qrCode.jpg', {  
  type: 'image/jpeg'  
});  
  
const dataTransfer = new DataTransfer();  
dataTransfer.items.add(finalFile);  
document.querySelector('input[type=file]').files = dataTransfer.files;
```

L'image conservée par le tag input est remplacée par le QRCode.