Nuez
Samuel
BTS SIO

La Sécurité Informatique

La sécurité informatique est un enjeu majeur en 2020. La plupart de nos infrastructures sont informatisées et susceptibles d'être piratés par des personnes mal intentionnées. Pour ce faire nous, développeurs et réseaux, devons travailler main dans la main afin de sécuriser nos applications et nos services.

Côté Développeurs :

Pour un développeur, il est important de se renseigner systématiquement sur les nouveautés d'un langage, d'une librairie ou autre… une méthode de **cryptage / hashage** peut faire son apparition et être plus sécurisée. Bien sûr avant de faire un code sécurisé, il faut bien le faire. Donc il est important dans un premier temps de bien nommer ses variables, **commenter son code** (c'est important, très important).

En effet le hashage, plus communément appelé cryptage, est une méthode qui permet de modifier une chaîne de caractère initiale (exemple : « impossible ») pour qu'elle ne soit pas lisible directement par le pirate. Il devra passer par une face de décryptage.
Suivant les méthodes employées, le hashage peut être difficile à casser. Il en existe plein, mais certains sont à utiliser plus que d'autres car ils peuvent être cassés en quelques secondes ou bien certains sont meilleurs que d'autre. Pour sécuriser d'avantage des données, un salage peut être fait. C'est à dire prendre une variable, un nom ou qu'importe, l'appliquer en tant que suffixe / préfixe à ce qu'on souhaite hasher.
En tant que développeur il est primordial de comprendre ce mécanisme car dans le cadre d'un site web ou bien d'une infrastructure pouvant regrouper une grosse quantité d'informations, il est intolérable d'avoir un mot de passe en clair (sans système de hashage décrit précédemment).

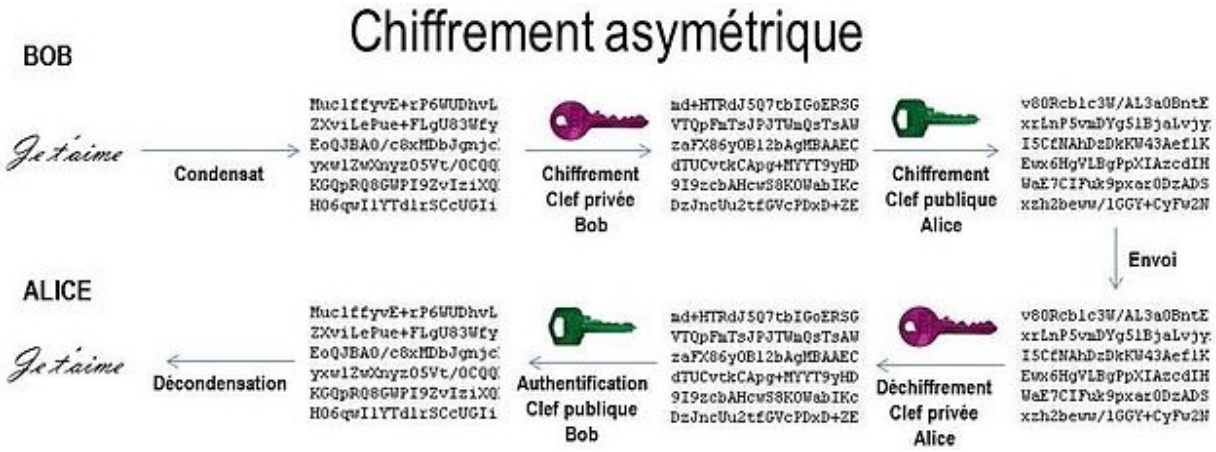
Pour sécuriser d'avantage son application, qu'elle soit web ou sur un poste physique (télécharger .msi / .exe), il est aussi préférable de **obfusquer** le nom de ses variables et de rendre son code illisible. En effet, cette méthode permet, dans le cas d'un **data mining**, de préserver son code source et de le rendre incompréhensible. Il est vital de faire cela car si le code source est compris de tous, il est évident que des bugs ou des manipulations peuvent êtes exploiter pour compromettre le travail que nous avons fait juste avant.
Le seul moment où un développeur peut se permettre de contourner cette règle est lorsqu'il publie son code sur github par exemple.

Par ailleurs, il est important de trouver les failles sur son application et de regarder constamment le CVE (***Common Vulnerabilities and Exposures).***

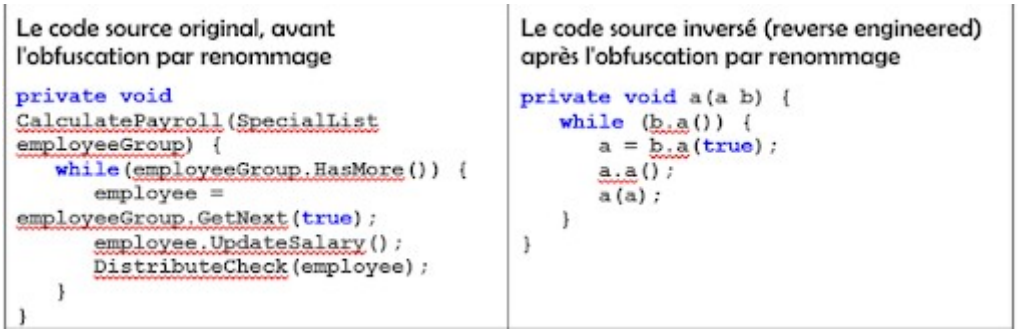
Côté Réseau :

Pour un réseau, il est, comme un développeur, important de se renseigner systématiquement sur les nouveautés que peut proposer un protocole, un outil ou autre. Par exemple il est important de se renseigner sur les features de Nginx ou Apache dans le cadre de l'hébergement d'un site web ou bien de WordPress et savoir s'il est bon de mettre à jour ou non cet outil-là. Le premier travail qu'un réseau doit faire est de comprendre le fonctionnement de sa machine, de l'optimiser et de vérifier s'il n'y aura pas de problème lors d'une mise à jour de paquets (donc avoir une partie prod et une partie test).
Dans le cadre de la sécurité informatique, la mise en place de protocole permettant de chiffrer les données reçues et émises avec un destinataire. Par exemple le **chiffrement RSA** peut être utilisé pour communiquer avec quelqu'un de façon sécurisée. Cet exemple peut être assimilé au certificat SSL (le petit cadenas vert sur les liens ou généralement le **https**).
Une des choses les plus importantes dans la sécurisation d'application que ce soit web ou physique, est d'empêcher coûte que coûte une personne à avoir accès au serveur dans son entièreté et il est important de sécuriser sa machine avec des mot de passe complexe (on évite le root root et autre variable pour s'authentifier en SSH) ou bien d'empêcher une connexion en root. On peut tout aussi faire une connexion via certificat ce qui est d'autant plus sécuriser car on bloque les connexions via mdp

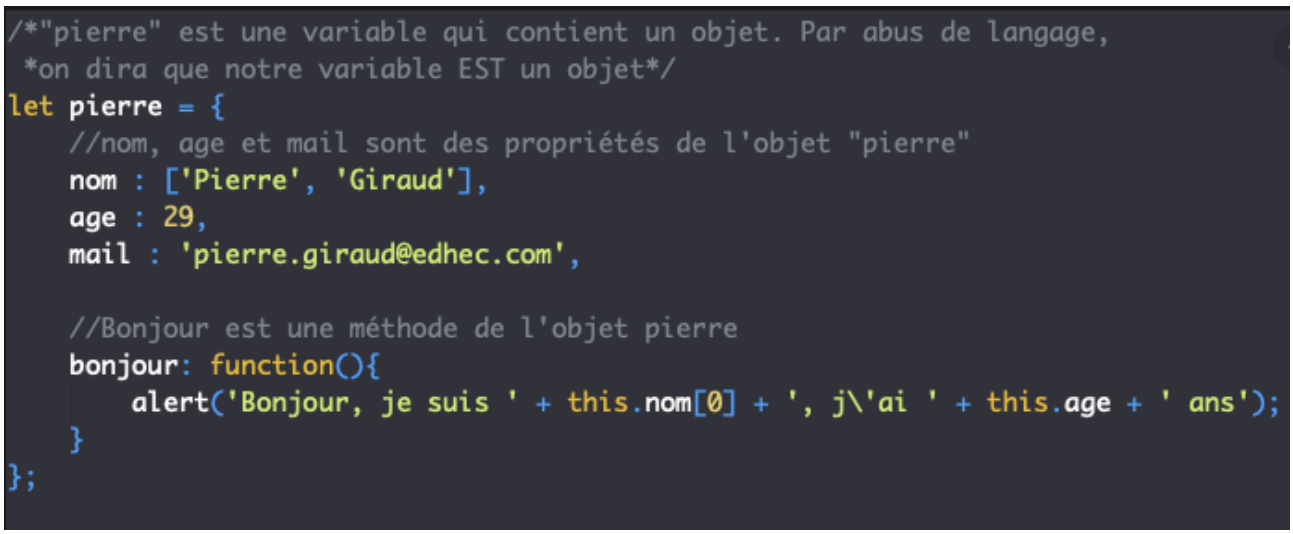
Quelques screenshot des bonnes pratiques :



chiffrement RSA → clé publique + clé privée donnant le même résultat



offuscation → on essaye de rendre son code illisible



```
/*"pierre" est une variable qui contient un objet. Par abus de langage,
 *on dira que notre variable EST un objet*/
let pierre = {
    //nom, age et mail sont des propriétés de l'objet "pierre"
    nom : ['Pierre', 'Giraud'],
    age : 29,
    mail : 'pierre.giraud@edhec.com',

    //Bonjour est une méthode de l'objet pierre
    bonjour: function(){
        alert('Bonjour, je suis ' + this.nom[0] + ', j\'ai ' + this.age + ' ans');
    }
};
```

nommage et mise en place de commentaires pour se repérer dans son code

Sources :

https://discord.gg/graven

https://openclassrooms.com/fr/courses/1733741-effectuez-votre-veille-en-cybersecurite

Utilisateur Discord :

TnTakara#2638
pilna#3151