LYCÉE ARISTIDE BERGÈS
SIO

Découverte

101 - Obfuscation de code JavaScript

Rédigé par

David ROUMANET
Professeur BTS SIO

Changement

Date	Révision

Sommaire

A Introduction

JavaScript s'exécute côté client, il est donc relativement facile d'avoir accès au code.

La solution pour sécuriser cette situation serait d'exécuter le code côté serveur, puis de renvoyer le résultat. Cela pose cependant deux problèmes :

- Le serveur serait facilement surchargé
- Toute action nécessiterait un échange réseau et donc une transaction dépendant de l'accès (de la fibre au GPRS sur téléphone)

Une autre solution consiste à rendre le code illisible, c'est le concept d'obscurcissement, que nous rencontrerons plus souvent sous l'anglicisme obfuscation.

A.1 Présentation

Cette exploration va permettre d'aborder la manière de générer un code illisible ou presque à un être humain, mais toujours exécutable par un interpréteur JavaScript.

Nous essayerons des méthodes manuelles, puis des systèmes automatisés.

A.2 Prérequis

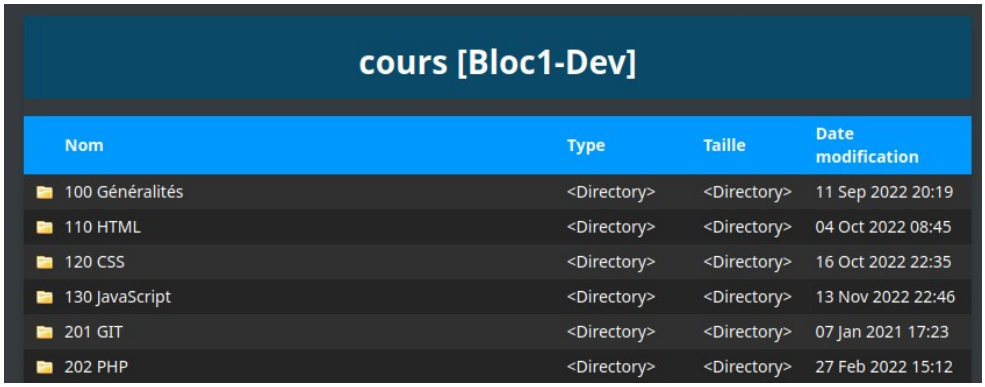
Connaître le langage JavaScript de manière avancée :

- les fonctions
- les conversions en ASCII, hexadécimale, binaire
- Les tableaux, les variables
- …

B Affichage de code

Tout d'abord, voici la technique pour afficher le code JavaScript présent sur un site.

Connectez-vous sur le site http://david.roumanet.free.fr/BTS-SIO/Bloc1-Dev/



cours [Bloc1-Dev]

Nom	Type	Taille	Date modification
100 Généralités	<Directory>	<Directory>	11 Sep 2022 20:19
110 HTML	<Directory>	<Directory>	04 Oct 2022 08:45
120 CSS	<Directory>	<Directory>	16 Oct 2022 22:35
130 JavaScript	<Directory>	<Directory>	13 Nov 2022 22:46
201 GIT	<Directory>	<Directory>	07 Jan 2021 17:23
202 PHP	<Directory>	<Directory>	27 Feb 2022 15:12

Le site utilise un programme en PHP pour afficher les fichiers et dossiers, mais utilise un script JavaScript pour les trier sur la page (cliquez sur 'Nom', 'Type', 'Taille' ou 'Date modification' pour le tester).

B.1 Utiliser le débogueur

Dans votre navigateur, utilisez la combinaison de touche `Ctrl`+`Shift`+`I` pour afficher le débogueur et…

- Cliquez sur l'onglet `Débogueur` (Firefox)
- Cliquez sur l'onglet `Sources` (Chrome)

Puis, cliquez sur le fichier `.sorttable.js` : le débogueur affiche le fichier. Vous devriez retrouver le nom de l'auteur de ce script version 2 datant de 2007.

C Obscurcissement simple

Il existe des outils pour masquer ou rendre moins lisible son code. Nous allons cependant étudier la manière dont c'est faisable "manuellement".

C.1 Affichage d'une phrase

Imaginons un code très simple :

```
// Instruction simple
console.log("LE CAPITAINE BOB A 56 ANS")
```

Transformer la fonction semble impossible pour le moment, mais on sait transformer une chaîne de caractères.

En utilisant l'astuce suivante, le code est plus long et moins lisible.

```
za=(String.fromCharCode(53)+String.fromCharCode(54))*1;ui49=String.fromCharCode(66,79,66);a47io=String.
fromCharCode(76,69,32,67,65,80,73,84,65,73,78,69,32);console.log(a47io+ui49+" A "+za+" ANS")
```

Les techniques utilisées ici sont :

- la **minification** (réduction des espaces, sauts de ligne et symboles inutiles)
- le **renommage** (nom de variables ou fonctions sans relation avec l'usage)
- le **découpage** (division de codes ou chaînes en plusieurs morceaux)
- l'**encodage** (chiffrement des informations)

Si vous testez ces deux codes, ils fonctionnent et s'affichent de manière identique, pourtant, l'un des deux a demandé plus de temps d'exécution pour recréer le fonctionnement normal.

Si nous replaçons les sauts de ligne, le code est déjà plus clair :

```
za=(String.fromCharCode(53)+String.fromCharCode(54))*1;
ui49=String.fromCharCode(66,79,66);
a47io=String.fromCharCode(76,69,32,67,65,80,73,84,65,73,78,69,32);
console.log(a47io+ui49+" A "+za+" ANS")
```

D Obscurcissement avancé

Dans ce cas, nous allons utiliser les fonctionnalités du site https://www.obfuscator.io/.

D.1 Source

Voici le code :

```
// Instruction simple
console.log("LE CAPITAINE BOB A 56 ANS")
```

D.2 Traitement

Collez le code dans le champ de saisie du site, puis cliquez sur la sélection [Option preset] et choisissez "Low".

D.3 Résultat

Voici le résultat du programme :

```
var _0x455390=_0x3358;(function(_0x11236b,_0x4a324c){var
_0x30b26f=_0x3358,_0x39e07c=_0x11236b();while(!![]){try{var _0xd779d2=parseInt(_0x30b26f(0x1b1))/0x1*(-
parseInt(_0x30b26f(0x1ac))/0x2)+-parseInt(_0x30b26f(0x1b3))/0x3+-parseInt(_0x30b26f(0x1c0))/0x4*(-
parseInt(_0x30b26f(0x1c2))/0x5)+parseInt(_0x30b26f(0x1b9))/0x6+-parseInt(_0x30b26f(0x1b2))/0x7*(-
parseInt(_0x30b26f(0x1be))/0x8)+-parseInt(_0x30b26f(0x1b8))/0x9+parseInt(_0x30b26f(0x1b7))/0xa*(-
parseInt(_0x30b26f(0x1ae))/0xb);if(_0xd779d2===_0x4a324c)break;else _0x39e07c['push']
(_0x39e07c['shift']());}catch(_0x126ef4){_0x39e07c['push'](_0x39e07c['shift']());}}}
(_0x1cec,0x4d435));function _0x3358(_0xe0faaa,_0x7da82f){var _0x43eec6=_0x1cec();return
_0x3358=function(_0x3de6f1,_0x5e6697){_0x3de6f1=_0x3de6f1-0x1ab;var
_0x2d45d9=_0x43eec6[_0x3de6f1];return _0x2d45d9;},_0x3358(_0xe0faaa,_0x7da82f);}var
_0xc66b90=(function(){var _0x5790f4=!![];return function(_0xe8e016,_0x1992d7){var _0x170758=_0x5790f4?
function(){var _0x920312=_0x3358;if(_0x1992d7){var _0x3cd9fd=_0x1992d7[_0x920312(0x1c3)]
(_0xe8e016,arguments);return _0x1992d7=null,_0x3cd9fd;}}:function(){};return _0x5790f4=!
[],_0x170758;};}()),_0x536cc2=_0xc66b90(this,function(){var _0x4ceeeb=_0x3358;return
_0x536cc2[_0x4ceeeb(0x1c1)]()['search']('(((.+)+)+)+$')[_0x4ceeeb(0x1c1)]()[_0x4ceeeb(0x1bb)]
(_0x536cc2)[_0x4ceeeb(0x1b4)](_0x4ceeeb(0x1bc));});_0x536cc2();var _0x5e6697=(function(){var
_0x1dad85=!![];return function(_0x1f19ad,_0x953cce){var _0x3f1192=_0x1dad85?function(){var
_0x18ffe9=_0x3358;if(_0x953cce){var _0x152448=_0x953cce[_0x18ffe9(0x1c3)](_0x1f19ad,arguments);return
_0x953cce=null,_0x152448;}}:function(){};return _0x1dad85=![],_0x3f1192;};}
()),_0x3de6f1=_0x5e6697(this,function(){var _0x4f04dd=_0x3358,_0x3ea4ea;try{var
_0xf19d6b=Function(_0x4f04dd(0x1b0)+_0x4f04dd(0x1b6)+');');_0x3ea4ea=_0xf19d6b();}catch(_0x1ad5ab)
{_0x3ea4ea=window;}var _0x1cf19b=_0x3ea4ea[_0x4f04dd(0x1ab)]=_0x3ea4ea[_0x4f04dd(0x1ab)]||
{},_0x15aeee=['log',_0x4f04dd(0x1c5),'info',_0x4f04dd(0x1bd),'exception','table','trace'];for(var
_0x7cd755=0x0;_0x7cd755<_0x15aeee[_0x4f04dd(0x1bf)];_0x7cd755++){var
_0x3a0378=_0x5e6697[_0x4f04dd(0x1bb)][_0x4f04dd(0x1ad)][_0x4f04dd(0x1af)]
(_0x5e6697),_0x53ad7a=_0x15aeee[_0x7cd755],_0x4fb2af=_0x1cf19b[_0x53ad7a]||
_0x3a0378;_0x3a0378[_0x4f04dd(0x1c4)]=_0x5e6697[_0x4f04dd(0x1af)]
(_0x5e6697),_0x3a0378[_0x4f04dd(0x1c1)]=_0x4fb2af[_0x4f04dd(0x1c1)]['bind']
(_0x4fb2af),_0x1cf19b[_0x53ad7a]=_0x3a0378;}});function _0x1cec(){var _0xe3b9b9=['return\
x20(function()\x20','2iEEcOz','7PZTkkU','7122gPGTzW','search','log','{}.constructor(\x22return\x20this\
x22)(\x20)','25370coEHvn','1319490VUZLPa','1149276PzzZbc','LE\x20CAPITAINE\x20BOB\x20A\x2056\
x20ANS','constructor','(((.+)+)+)+
$','error','4700912GEANCy','length','30408dBCakf','toString','310AZPJnT','apply','__proto__','warn','co
nsole','148244YVWmhF','prototype','2761sSgUcr','bind'];_0x1cec=function(){return _0xe3b9b9;};return
_0x1cec();}_0x3de6f1(),console[_0x455390(0x1b5)](_0x455390(0x1ba));
```

Vous pouvez tester ce code dans une page web, entre deux balises `<script>` et `</script>`.

E Éclaircir un code

Après un tel traitement, il convient de vérifier si l'opération inverse est possible.

Testons le code précédemment généré dans https://beautifier.io/

```
var _0x455390 = _0x3358;
(function(_0x11236b, _0x4a324c) {
    var _0x30b26f = _0x3358,
        _0x39e07c = _0x11236b();
    while (!![]) {
        try {
            var _0xd779d2 = parseInt(_0x30b26f(0x1b1)) / 0x1 * (-parseInt(_0x30b26f(0x1ac)) / 0x2) + -parseInt(_0x30b26f(0x1b3)) /
0x3 + -parseInt(_0x30b26f(0x1c0)) / 0x4 * (-parseInt(_0x30b26f(0x1c2)) / 0x5) + parseInt(_0x30b26f(0x1b9)) / 0x6 + -
parseInt(_0x30b26f(0x1b2)) / 0x7 * (-parseInt(_0x30b26f(0x1be)) / 0x8) + -parseInt(_0x30b26f(0x1b8)) / 0x9 +
parseInt(_0x30b26f(0x1b7)) / 0xa * (-parseInt(_0x30b26f(0x1ae)) / 0xb);
            if (_0xd779d2 === _0x4a324c) break;
            else _0x39e07c['push'](_0x39e07c['shift']());
        } catch (_0x126ef4) {
            _0x39e07c['push'](_0x39e07c['shift']());
        }
    }
}(_0x1cec, 0x4d435));

function _0x3358(_0xe0faaa, _0x7da82f) {
    var _0x43eec6 = _0x1cec();
    return _0x3358 = function(_0x3de6f1, _0x5e6697) {
        _0x3de6f1 = _0x3de6f1 - 0x1ab;
        var _0x2d45d9 = _0x43eec6[_0x3de6f1];
        return _0x2d45d9;
    }, _0x3358(_0xe0faaa, _0x7da82f);
}
var _0xc66b90 = (function() {
        var _0x5790f4 = !![];
        return function(_0xe8e016, _0x1992d7) {
            var _0x170758 = _0x5790f4 ? function() {
                var _0x920312 = _0x3358;
                if (_0x1992d7) {
                    var _0x3cd9fd = _0x1992d7[_0x920312(0x1c3)](_0xe8e016, arguments);
                    return _0x1992d7 = null, _0x3cd9fd;
                }
            } : function() {};
            return _0x5790f4 = ![], _0x170758;
        };
    }()),
    _0x536cc2 = _0xc66b90(this, function() {
        var _0x4ceeeb = _0x3358;
        return _0x536cc2[_0x4ceeeb(0x1c1)]()['search']('(((.+)+)+)+$')[_0x4ceeeb(0x1c1)]()[_0x4ceeeb(0x1bb)](_0x536cc2)
[_0x4ceeeb(0x1b4)](_0x4ceeeb(0x1bc));
    });
_0x536cc2();
var _0x5e6697 = (function() {
        var _0x1dad85 = !![];
        return function(_0x1f19ad, _0x953cce) {
            var _0x3f1192 = _0x1dad85 ? function() {
                var _0x18ffe9 = _0x3358;
                if (_0x953cce) {
                    var _0x152448 = _0x953cce[_0x18ffe9(0x1c3)](_0x1f19ad, arguments);
                    return _0x953cce = null, _0x152448;
                }
            } : function() {};
            return _0x1dad85 = ![], _0x3f1192;
        };
    }()),
    _0x3de6f1 = _0x5e6697(this, function() {
        var _0x4f04dd = _0x3358,
            _0x3ea4ea;
        try {
            var _0xf19d6b = Function(_0x4f04dd(0x1b0) + _0x4f04dd(0x1b6) + ');');
            _0x3ea4ea = _0xf19d6b();
        } catch (_0x1ad5ab) {
            _0x3ea4ea = window;
        }
        var _0x1cf19b = _0x3ea4ea[_0x4f04dd(0x1ab)] = _0x3ea4ea[_0x4f04dd(0x1ab)] || {},
            _0x15aeee = ['log', _0x4f04dd(0x1c5), 'info', _0x4f04dd(0x1bd), 'exception', 'table', 'trace'];
        for (var _0x7cd755 = 0x0; _0x7cd755 < _0x15aeee[_0x4f04dd(0x1bf)]; _0x7cd755++) {
            var _0x3a0378 = _0x5e6697[_0x4f04dd(0x1bb)][_0x4f04dd(0x1ad)][_0x4f04dd(0x1af)](_0x5e6697),
                _0x53ad7a = _0x15aeee[_0x7cd755],
                _0x4fb2af = _0x1cf19b[_0x53ad7a] || _0x3a0378;
            _0x3a0378[_0x4f04dd(0x1c4)] = _0x5e6697[_0x4f04dd(0x1af)](_0x5e6697), _0x3a0378[_0x4f04dd(0x1c1)] =
_0x4fb2af[_0x4f04dd(0x1c1)]['bind'](_0x4fb2af), _0x1cf19b[_0x53ad7a] = _0x3a0378;
        }
    });

function _0x1cec() {
```

```
    var _0xe3b9b9 = ['return\x20(function()\x20', '2iEEcOz', '7PZTkkU', '7122gPGTzW', 'search', 'log', '{}.constructor(\x22return\
x20this\x22)(\x20)', '25370coEHvn', '1319490VUZLPa', '1149276PzzZbc', 'LE\x20CAPITAINE\x20BOB\x20A\x2056\x20ANS', 'constructor',
'(((.+)+)+)+$', 'error', '4700912GEANCy', 'length', '30408dBCakf', 'toString', '310AZPJnT', 'apply', '__proto__', 'warn', 'console',
'148244YVWmhF', 'prototype', '2761sSgUcr', 'bind'];
    _0x1cec = function() {
        return _0xe3b9b9;
    };
    return _0x1cec();
}
_0x3de6f1(), console[_0x455390(0x1b5)](_0x455390(0x1ba));
```

Cette fois, le code reste pratiquement illisible : on peut retrouver une trace de la chaîne originale et le mot console qui peuvent nous mettre sur la voix mais cela reste insuffisant pour une personne ne connaissant pas le code d'origine.

Vous pouvez relancer une 'obfuscation' en utilisant [Option preset] sur 'High'.

F Conclusion

Cette technique n'est pas parfaite, mais peut éviter de révéler un algorithme innovant ou critique pour l'organisation. Il est important de conserver le code source dans un endroit sécurisé, car il ne sera pas possible d'apporter de modification sur le code obscurci.

Cette technique ne doit servir que sur des bouts de codes critiques : l'occupation en mémoire et les pertes de performances sont en effet décuplés. Il revient au développeur de prendre la décision d'utiliser cette méthode, ou de choisir un traitement côté serveur.