

## Installation de Fiddler

### 1. Télécharger Fiddler:

- Allez sur le site officiel de Fiddler: <https://www.telerik.com/download/fiddler>
- Téléchargez la version de Fiddler pour votre système d'exploitation (Windows).

### 2. Installer Fiddler:

- Exécutez le fichier d'installation téléchargé.
- Suivez les instructions à l'écran pour compléter l'installation.

## Configuration de Fiddler pour Intercepter le Trafic HTTP et HTTPS

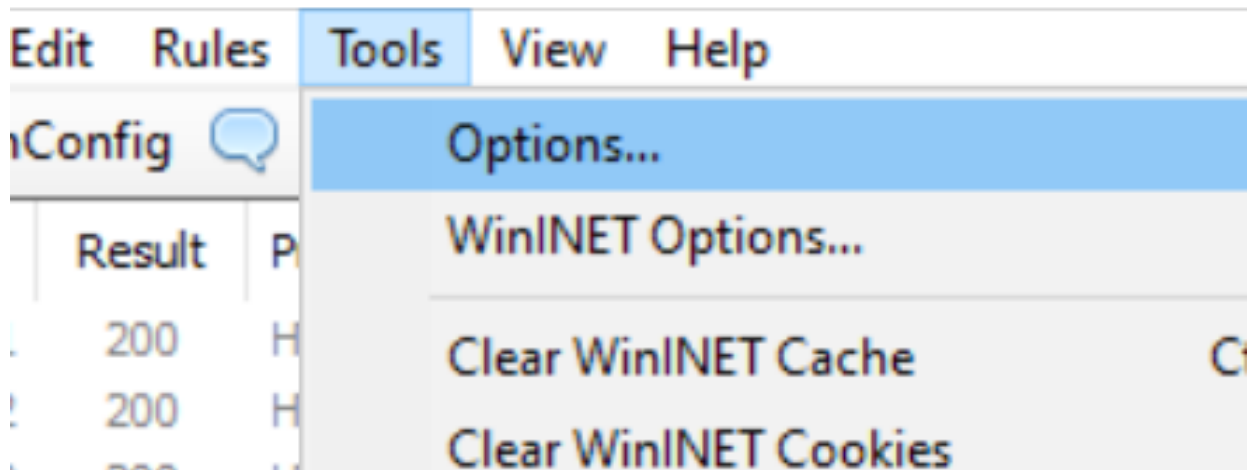
### 1. Lancer Fiddler:

- Ouvrez Fiddler après l'installation.

### 2. Configurer la capture HTTPS:

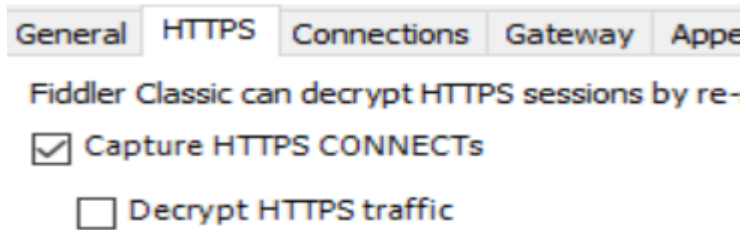
- Allez dans **Tools > Options**.

Progress Telerik Fiddler Classic



- Sélectionnez l'onglet **HTTPS**.

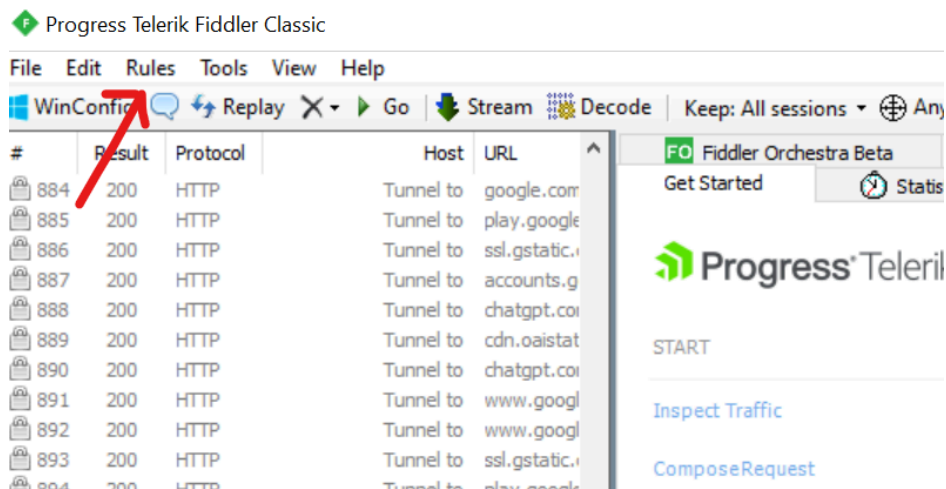
Options

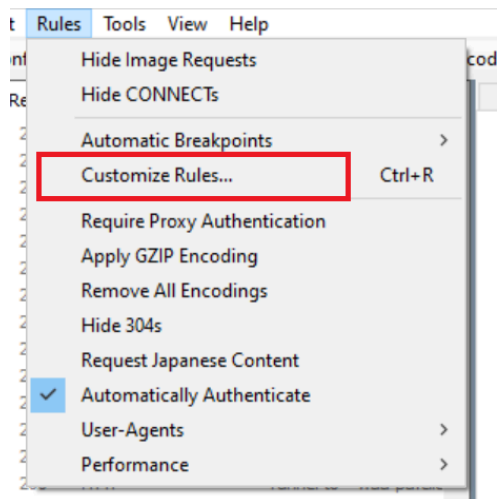
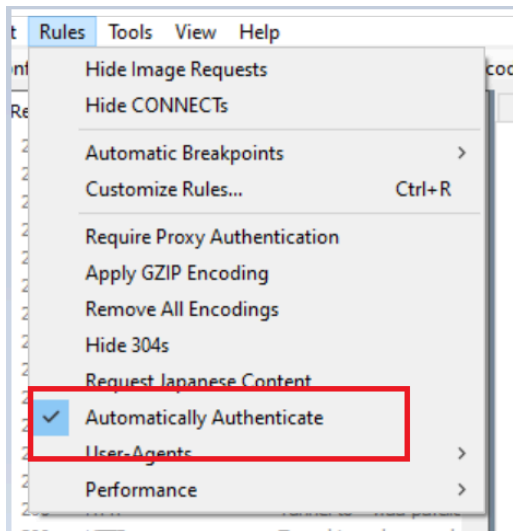


## Modification des Requêtes HTTP et HTTPS

### 1. Utiliser les Rules :

- Allez dans **Rules > Customize Rules**.
- Cela ouvrira un éditeur de texte où vous pourrez écrire des scripts en utilisant JScript.NET pour modifier les requêtes et réponses.





## ajouter func getMACAddress() et leur contenu à l'intérieur de class Handlers

```
class Handlers
{
    static function getMACAddress(): String
    {
        var macAddress: String = "";

        // Récupérer toutes les interfaces réseau
        var nics: NetworkInterface[] = NetworkInterface.GetAllNetworkInterfaces();

        // Parcourir les interfaces pour trouver la première active non-loopback et non-virtuelle
        for (var i: int = 0; i < nics.Length; i++)
        {
            var nic = nics[i];
            if (nic.OperationalStatus == OperationalStatus.Up &&
                nic.NetworkInterfaceType != NetworkInterfaceType.Loopback &&
                nic.NetworkInterfaceType != NetworkInterfaceType.Tunnel &&
                !nic.Description.Contains("Virtual") &&
                !nic.Description.Contains("Pseudo"))
            {
                macAddress = nic.GetPhysicalAddress().ToString();
                if (!String.IsNullOrEmpty(macAddress))
                {
                    break;
                }
            }
        }

        // Si aucune adresse MAC valide n'est trouvée, lever une exception
        if (String.IsNullOrEmpty(macAddress))
        {
            macAddress = "00:00:00:00:00:00";
        }

        return macAddress;
    }
}
```

## Chercher sur la fonction OnBeforeRequest et ajouter ce script:

```
static function OnBeforeRequest(oSession: Session) {
    // Sample Rule: Color ASPX requests in RED
    // if (oSession.uriContains(".aspx")) { oSession["ui-color"] = "red"; }

    // Sample Rule: Flag POSTs to fiddler2.com in italics
    // if (oSession.HostnameIs("www.fiddler2.com") && oSession.HTTPMethodIs("POST")) { oSession["ui-italic"] = "yup"; }

    // Sample Rule: Break requests for URLs containing "/sandbox/"
    // if (oSession.uriContains("/sandbox/")) {
    //     oSession.oFlags["x-breakrequest"] = "yup"; // Existence of the x-breakrequest flag creates a breakpoint;
    // }
}
```

```
var macAddress = getMACAddress();

oSession.oRequest.headers.Add("mac-adresse", macAddress);
|
```

enregistrer les modifications.

**Les scripts seront trouvés dans le même dossier que ce fichier. Vous pouvez les trouver là .**