

Содержание

[Вопрос:](#)

Вопрос:

Что такое файл Pac Прокси и там какие-либо примеры?

Среда: Cisco Web Security Appliance

Примечание: Эта статья Базы Знаний ссылается на программное обеспечение, которое не поддерживается Cisco. Информация предоставлена как любезность для вашего удобства. Для дальнейшей поддержки свяжитесь с поставщиком программного обеспечения.

Опорный канал: [JavaScript или образцы файла автопрокси JScript](#)

JavaScript или образцы файла автопрокси JScript

Следующие сценарии предоставляют примеры того, как '.pac' файл мог использоваться для определения URL автопрокси. Для использования этих функций необходимо поменять имена прокси, номера портов и IP-адреса.

1. `isInNet`, `isResolvable`, и функции `dnsResolve` сделали запрос сервера DNS
2. Ссылки на объекты Объектной модели, свойства или методы заставляют файл автоматической конфигурации прокси отказывать тихо. Например, ссылки `window.open (... .alert...)`, и `пароль (...)` вся причина файл автоматической конфигурации прокси для сбоя на Internet Explorer.

Пример 1: Прямое подключение локальных хостов, все другие соединяются через прокси

Следующая функция проверяет, чтобы видеть, является ли имя хоста локальным хостом, и если это, является ли соединение прямым. Если имя хоста не является локальным хостом, соединение через прокси (прокси).

функция `FindProxyForURL` (URL, хост)

```
{  
если (isPlainHostName (хост))  
возвратите "ПРЯМОЙ";  
еще  
return "PROXY proxy:80";  
}
```

`isPlainHostName` функционируют проверки, чтобы видеть, существуют ли какие-либо точки в имени хоста. Если так, это возвращает `false`; иначе, функция возвращает `true`.

Примечание: Для прямого подключения к локальному веб-серверу могли бы также быть необходимы домен или соответствие подсети. Не все запросы к локальным веб-серверам сделаны с помощью просто имени хоста

Пример 2: Хосты в подключении межсетевого экрана прямые, внешние локальные серверы соединяются через прокси

Следующая функция проверяет, чтобы видеть, является ли хост любой "простым" именем хоста (значение, что доменное имя не включено), или часть отдельного домена (.company.com), но именем хоста не является или www или домой.

функция FindProxyForURL (URL, хост)

```
{
если ((isPlainHostName (хост) ||
dnsDomainIs (хост, ".company.com")) &&
! localHostOrDomainIs (хост, "www.company.com") &&
! localHostOrDoaminIs (хост, "home.company.com"))
возвратите "ПРЯМОЙ";
еще
return "PROXY proxy:80";
}
```

Примечание:

- Функция **localHostOrDomainIs** выполняется только для URL в локальном домене.
- Если домен имени хоста совпадает с данным доменом, функция **dnsDomainIs** возвращает true.

Пример 3: Если хост разрешим, прямое подключение. В противном случае подключение с помощью прокси.

Следующая функция просит, чтобы сервер DNS попытался решить, что имя хоста прошло к нему. Если это может, то прямое подключение сделано. Если это не может, соединение, которое будет делаться через прокси. Когда внутренний сервер DNS используется для решения всех внутренних имен хоста, это полезно.

функция FindProxyForURL (URL, хост)

```
{
если (isResolvable (хост))
возвратите "ПРЯМОЙ";
еще
return "PROXY proxy:80";
}
```

Посмотрите примечание по функции isResolvable в верхней части страницы.

Пример 4: Если хост находится в указанной подсети, прямое подключение. В противном случае подключение с помощью прокси.

Следующая функция сравнивает данный образец IP-адреса и маску с именем хоста. Это полезно, если определенные хосты в подсети должны быть связаны непосредственно, и другие должны быть связаны с помощью прокси.

функция FindProxyForURL (URL, хост)

```
{
если (isInNet (хост, "999.99.9.9", "255.0.255.0"))
возвратите "ПРЯМОЙ";
еще
```

```
return "PROXY proxy:80";
}
```

Посмотрите примечание по функции **isInNet** в верхней части страницы.

Если адрес IP - адреса хоста совпадает с указанным образцом, **isInNet** (хост, образец, маска) функция возвращает true. Маска указывает который часть IP-адреса совпасть (255=match, 0=ignore).

Пример 5: Определите тип соединения на основе домена хоста

Если хост локален, следующая функция задает прямое подключение. Если хост не локален, эта функция определяет который прокси использовать на основе домена хоста. Если доменное имя хоста является одним из критериев для выбора прокси, это полезно.

функция FindProxyForURL (URL, хост)

```
{
если (isPlainHostName (хост))
возвратите "ПРЯМОЙ";
еще, если (shExpMatch (хост, "*.com"))
возвратите "ПРОКСИ comproxy:80";
еще, если (shExpMatch (хост, "*.edu"))
возвратите "ПРОКСИ eduproxy:80";
еще
возвратите "прокси ПРОКСИ";
}
```

Если str совпадает с shexp, использующим характер экспрессии оболочки, **shExpMatch** (str, shexp) функция возвращает true.

Пример 6: Определите тип соединения на основе используемого протокола

Следующая функция извлекает используемый протокол и делает выбор прокси соответственно. Если никакое соответствие не сделано на протоколе, то прямое подключение сделано. Если используемый протокол является одним из критериев для выбора прокси, это полезно.

функция FindProxyForURL (URL, хост)

```
{
если (url.substring (0, 5) == "http:") {
return "PROXY proxy:80";
}
еще, если (url.substring (0, 4) == "ftp:") {
возвратите "ПРОКСИ fproxy:80";
}
еще, если (url.substring (0, 7) == "gopher:") {
возвратите "ПРОКСИ gproxy";
}
еще, если (url.substring (0, 6) == "https:") {
возвратите "ПРОКСИ secproxy:8080";
}
еще {
возвратите "ПРЯМОЙ";
}
```

```
}  
}
```

Функция подстроки извлекает заданный номер символов от строки.

Пример 7: Определите параметр прокси путем проверки, чтобы видеть, совпадает ли имя хоста с IP-адресом

Следующая функция делает выбор прокси путем перевода имени хоста в IP-адрес и сравнения его с заданной строкой.

функция FindProxyForURL (URL, хост)

```
{  
если (dnsResolve (хост) == "999.99.99.999") {/= http://прокси  
возвратите "ПРОКСИ secproxy:8080";  
}  
еще {  
return "PROXY proxy:80";  
}  
}
```

Посмотрите примечание по функции **dnsResolve** в верхней части страницы.

Пример 8: Если соответствия IP - адреса хоста указанный IP, подключение через прокси, еще соединяются прямой

Следующая функция является другим способом сделать выбор прокси на основе определения IP-адреса. Данный пример, в отличие от примера 7, использует вызов функции для явного получения числового IP-адреса (Пример 7 использования функция **dnsResolve** для перевода имени хоста в числовой IP-адрес).

функция FindProxyForURL (URL, хост)

```
{  
если (myIpAddress () == "999.99.999.99") {  
return "PROXY proxy:80";  
}  
еще {  
возвратите "ПРЯМОЙ";  
}  
}
```

myIpAddress функция возвращает IP-адрес (в целочисленно-точечном формате) хоста, на котором работает браузер.

Пример 9: Если существуют какие-либо точки в имени хоста, подключение с помощью прокси. В противном случае, прямое подключение.

Следующая функция проверяет для наблюдения, сколько точек находится в имени хоста. Если существуют какие-либо точки в имени хоста, сделайте соединение через прокси. Если нет никаких точек в имени хоста, сделайте прямое подключение. Это - другой способ определить типы соединения на основе характеристик имени хоста.

функция FindProxyForURL (URL, хост)

```
{  
если (dnsDomainLevels (хост)> 0) {/, если количество точек в хосте> 0
```

```
return "PROXY proxy:80";
}
возвратите "ПРЯМОЙ";
}
```

dnsDomainLevels функция возвращает целое число, равное количеству точек в имени хоста.

Пример 10: Задайте дни недели для соединения через прокси, другое дневное прямое подключение

Следующая функция определяет тип соединения путем определения дней недели, которые являются соответствующими прокси. Дни, которые не падают между этими параметрами, используют прямое подключение. Эта функция могла быть полезной в ситуациях, где вы могли бы хотеть использовать прокси, когда трафик тяжел, и позвольте прямое подключение, когда трафик является светом.

функция FindProxyForURL (URL, хост)

```
{
если (weekdayRange (в "среду", "НАХОДИЛСЯ", "GMT")),
return "PROXY proxy:80";
еще
возвратите "ПРЯМОЙ";
}
```

weekdayRange (day1 [day2] [GMT]) функция возвращается, находится ли текущее системное время в пределах диапазона, заданного параметрами day1, day2, и GMT, Только первый параметр требуется. GMT параметр предполагает, что значения времени находятся во Время по Гринвичу, а не местный часовой пояс.

Пример 11: Распределение нагрузки с функцией простых вычислений

функция FindProxyForURL (URL, хост) {

возвратите randomProxy ();

}

функционируйте randomProxy ()

{

коммутатор (Math.floor (Math.random () *2))

{

случай 0: возвратите "ПРОКСИ 1.1.1.1:3128; ПРОКСИ 2.2.2.2; ПРЯМОЙ ПРОКСИ";

пример 1: возвратите "ПРОКСИ 2.2.2.2:3128; ПРОКСИ 1.1.1.1; ПРЯМОЙ ПРОКСИ";

}

}

Вы будете видеть, что случаи также внедряют аварийное переключение в случае, если один прокси не доступен.