

# Cos'è il file Proxy Pac e sono disponibili alcuni esempi?

## Sommario

[Domanda:](#)

## Domanda:

Che cos'è un file Proxy Pac e quali esempi?

**Ambiente:** Cisco Web Security Appliance

**Nota:** Questo articolo della Knowledge Base fa riferimento a software che non è gestito o supportato da Cisco. Le informazioni vengono fornite per comodità dell'utente. Per ulteriore assistenza, contattare il fornitore del software.

Link di riferimento: [File di esempio JavaScript o JScript Auto-Proxy](#)

## File di esempio JavaScript o JScript Auto-Proxy

I seguenti script spiegano come utilizzare un file ".pac" per specificare un URL auto-proxy. Per usare queste funzionalità, devi modificare i nomi, i numeri di porta e gli indirizzi IP del proxy.

1. Le funzioni **isInNet**, **isResolvable** e **dnsResolve** interrogano un server DNS
2. I riferimenti agli oggetti, alle proprietà o ai metodi del modello a oggetti generano automaticamente errori nel file di configurazione automatica proxy. Ad esempio, i riferimenti **window.open(...)**, **alert(...)** e **password(...)** generano tutti errori nel file di configurazione automatica proxy in Internet Explorer.

### Esempio 1: Gli host locali si connettono direttamente, tutti gli altri tramite proxy

La funzione seguente verifica se il nome host è un host locale e, in caso affermativo, se la connessione è diretta. Se il nome host non è un host locale, la connessione avviene tramite il proxy (proxy).

funzione FindProxyForURL(url, host)

```
{  
if (isPlainHostName(host))  
restituire "DIRETTAMENTE";  
Altrimenti  
restituire "proxy:80";  
}
```

La funzione **isPlainHostName** verifica la presenza di punti nel nome host. In caso affermativo,

restituisce false; in caso contrario, la funzione restituisce true.

**Nota:** Per una connessione diretta al server Web locale, potrebbe essere necessaria una corrispondenza di dominio o subnet. Non tutte le richieste ai server Web locali vengono eseguite utilizzando solo i nomi host

### Esempio 2: Gli host all'interno del firewall si connettono direttamente, mentre i server locali esterni si connettono tramite proxy

La funzione seguente verifica se l'host è un nome host "normale" (ovvero se il nome di dominio non è incluso) o parte di un particolare dominio (.company.com) ma il nome host non è www o home.

```
funzione FindProxyForURL(url, host)
{
  if (isPlainHostName(host) ||
      dnsDomainIs(host, ".company.com") &&
      !localHostOrDomainIs(host, "www.company.com") &&
      !localHostOrDomainIs(host, "home.azienda.com"))
    restituire "DIRETTAMENTE";
  Altrimenti
    restituire "proxy:80";
}
```

#### Nota:

- La funzione **localHostOrDomainIs** viene eseguita solo per gli URL nel dominio locale.
- La funzione **dnsDomainIs** restituisce true se il dominio del nome host corrisponde al dominio specificato.

### Esempio 3: Se l'host è risolvibile, connettersi direttamente. In caso contrario, connettersi utilizzando un proxy.

La funzione seguente richiede al server DNS di tentare di risolvere il nome host passato. Se possibile, viene stabilita una connessione diretta. In caso contrario, la connessione viene effettuata tramite proxy. Ciò è utile quando si utilizza un server DNS interno per risolvere tutti i nomi host interni.

```
funzione FindProxyForURL(url, host)
{
  if (isResolvable(host))
    restituire "DIRETTAMENTE";
  Altrimenti
    restituire "proxy:80";
}
```

Vedere la nota sulla funzione isResolvable nella parte superiore della pagina.

### Esempio 4: Se l'host si trova nella subnet specificata, connettersi direttamente. In caso contrario, connettersi utilizzando un proxy.

La funzione seguente confronta un modello e una maschera di indirizzo IP specifici con il nome

host. Ciò è utile se alcuni host di una subnet devono essere connessi direttamente e altri devono essere connessi tramite un proxy.

```
funzione FindProxyForURL(url, host)
{
  if (isInNet(host, "999.99.9", "255.0.255.0"))
    restituire "DIRETTAMENTE";
  Altrimenti
    restituire "proxy:80";
}
```

Vedere la nota sulla funzione **isInNet** nella parte superiore della pagina.

La funzione **isInNet**(host, modello, maschera) restituisce true se l'indirizzo IP dell'host corrisponde al modello specificato. La maschera indica quale parte dell'indirizzo IP deve corrispondere (255=corrispondenza, 0=ignora).

### Esempio 5: Determinare il tipo di connessione in base al dominio host

La funzione seguente specifica una connessione diretta se l'host è locale. Se l'host non è locale, questa funzione determina quale proxy utilizzare in base al dominio host. Ciò è utile se il nome di dominio host è uno dei criteri per la selezione del proxy.

```
funzione FindProxyForURL(url, host)
{
  if (isPlainHostName(host))
    restituire "DIRETTAMENTE";
  else if (shExpMatch(host, "*.com")
    restituire "PROXY COMPROXY:80";
  else if (shExpMatch(host, "*.edu")
    restituire "PROXY eduproxy:80";
  Altrimenti
    restituire "proxy";
}
```

La funzione **shExpMatch**(str, shexp) restituisce true se str corrisponde a shexp utilizzando modelli di espressioni della shell.

### Esempio 6: Determinare il tipo di connessione in base al protocollo utilizzato

La funzione seguente estrae il protocollo in uso ed effettua di conseguenza la selezione del proxy. Se il protocollo non corrisponde, viene stabilita una connessione diretta. Questa opzione è utile se il protocollo utilizzato è uno dei criteri per la selezione del proxy.

```
funzione FindProxyForURL(url, host)
{
  if (url.substring(0, 5) == "http:") {
    restituire "proxy:80";
  }
  else if (url.substring(0, 4) == "ftp:") {
    restituire "PROXY fproxy:80";
  }
  else if (url.substring(0, 7) == "gopher:") {
    restituire "PROXY";
  }
}
```

```

}
else if (url.substring(0, 6) == "https:") {
  restituire "PROXY secproxy:8080";
}
else {
  restituire "DIRETTAMENTE";
}
}
}

```

La funzione substring estrae il numero di caratteri specificato da una stringa.

### Esempio 7: Determinare l'impostazione proxy controllando se il nome host corrisponde all'indirizzo IP

La funzione seguente effettua una selezione proxy convertendo il nome host in un indirizzo IP e confrontandolo con una stringa specificata.

```

funzione FindProxyForURL(url, host)
{
  if (dnsResolve(host) == "999.99.99.999") { // = http://proxy
    restituire "PROXY secproxy:8080";
  }
  else {
    restituire "proxy:80";
  }
}

```

Vedere la nota sulla funzione **dnsResolve** nella parte superiore della pagina.

### Esempio 8: Se l'indirizzo IP dell'host corrisponde all'indirizzo IP specificato, connettersi tramite proxy, altrimenti connettersi direttamente

La seguente funzione è un altro modo per effettuare una selezione proxy basata sulla specifica di un indirizzo IP. In questo esempio, a differenza dell'esempio 7, viene utilizzata la chiamata di funzione per ottenere in modo esplicito l'indirizzo IP numerico. Nell'esempio 7 viene utilizzata la funzione **dnsResolve** per convertire il nome host nell'indirizzo IP numerico.

```

funzione FindProxyForURL(url, host)
{
  if (indirizzoIP() == "999.99.999.999") {
    restituire "proxy:80";
  }
  else {
    restituire "DIRETTAMENTE";
  }
}

```

La funzione **myIpAddress** restituisce l'indirizzo IP (in formato punto intero) dell'host su cui è in esecuzione il browser.

### Esempio 9: Se il nome host contiene punti, connettersi utilizzando un proxy. In caso contrario, effettuare la connessione direttamente.

La funzione seguente verifica il numero di punti presenti nel nome host. Se il nome host contiene punti, effettuare una connessione tramite proxy. Se il nome host non contiene punti, stabilire una connessione diretta. Questo è un altro modo per determinare i tipi di connessione in base alle caratteristiche del nome host.

```
funzione FindProxyForURL(url, host)
{
  if (dnsDomainLevels(host) > 0) { // se il numero di punti in host > 0
    restituire "proxy:80";
  }
  restituire "DIRETTAMENTE";
}
```

La funzione **dnsDomainLevels** restituisce un intero uguale al numero di punti nel nome host.

### Esempio 10: Specificare i giorni della settimana per la connessione tramite proxy, altri giorni per la connessione diretta

La funzione seguente determina il tipo di connessione specificando i giorni della settimana appropriati per un proxy. I giorni che non rientrano tra questi parametri utilizzano una connessione diretta. Questa funzione può essere utile nelle situazioni in cui si desidera utilizzare un proxy quando il traffico è pesante e consentire una connessione diretta quando il traffico è leggero.

```
funzione FindProxyForURL(url, host)
{
  if(weekDayRange("MER", "SAT", "GMT"))
    restituire "proxy:80";
  Altrimenti
    restituire "DIRETTAMENTE";
}
```

La funzione **weekDayRange**(day1 [,day2] [,GMT] ) restituisce se l'ora di sistema corrente rientra nell'intervallo specificato dai parametri day1, day2 e GMT. È necessario specificare solo il primo parametro. Il parametro GMT presume che i valori dell'ora siano espressi nell'ora di Greenwich anziché nel fuso orario locale.

### Esempio 11: Bilanciamento del carico con una semplice funzione matematica

```
funzione FindProxyForURL(url, host){
  return randomProxy();
}
funzione randomProxy()
{
  switch( Math.floor( Math.random() *2))
  {
  caso 0: restituire "PROXY 1.1.1.1:3128; PROXY 2.2.2.2; PROXY DIRECT;"
  caso 1: restituire "PROXY 2.2.2.2:3128; PROXY 1.1.1.1; PROXY DIRECT;"
  }
}
```

In questi casi viene inoltre implementato un failover nel caso in cui un proxy non sia disponibile.