

Forward Proxies

General forward proxies

- Operating
 - Proxy PAC Files

Operating

Proxy PAC Files

Matching in proxy.pac

Problem

The inclusion of a leading period (.) within the “host” parameter’s value seems to be ignored. In the example below, this causes “fakewebex(dot)com and “fake(dot)webex(dot)com” to both match the shExpMatch, which is not ideal.

```
function FindProxyForURL(url, host) {  
  if (shExpMatch(host, "*.webex.com"))  
    return "PROXY ${ZAPP_TUNNEL2_BYPASS}";  
  return "DIRECT";  
}
```

Solution

“ shExpMatch is a regular expression match function. RegEx considers a dot as matching any character - so webex.com would match “webex.com” as well as “webexxcom” and webex1com”.

You should consider

```
dnsDomainis(Host,"webex.com"
```

as a function, which is more efficient, but also more specific.

If you need more specificity using RegEx then

```
shExpMatch(host,".*webex.com")
```

would be a more appropriate regular expression, since the dot is escaped - meaning it should explicitly match the dot. This function is essentially the same

as the `dnsDomains` function above, but is still using RegEx which makes it computationally less efficient (not that it particularly matters)

The **shExpMatch** example `shExpMatch(host, ".*webex.com")` does indeed cause `fakewebex.com` to no longer get matched, and so it meets the goal. The code below that meets all of the criteria (matches on `webex.com` and `anything.webex.com`, but doesn't match on `fakewebex.com`, or `fakewebexcom`).

```
function FindProxyForURL(url, host) {  
  if (dnsDomains(host, "webex.com") || dnsDomains(host, ".webex.com"))  
    return "PROXY ${ZAPP_TUNNEL2_BYPASS}";  
  return "DIRECT";  
}
```

Source: <https://community.zscaler.com/t/pac-file-syntax-shexpmatch-period-seemingly-ignored/16705/4>

Links

Link	Description
http://findproxyforurl.com/pac-functions/	PAC file functions and more information
https://thorsen.pm/	Online proxy PAC file tester