

Proxy サーバ (Squid) の使用方法について

2021年7月初版

FITELnet F70/F71/F220/F221 (以下、本装置) で Squid を使ってコンテナ上に proxy サーバを構築する方法について本紙でご説明します。

※文中で引用している資料は下記 URL に掲載されています。

<https://www.furukawa.co.jp/fitelnet/product/container/lxc/index.html>

下記、図 1 のネットワーク構成において、コンテナのインタフェース eth1 は装置内部の L2 スイッチを介して(\*1)インタフェース Giga1/1 に接続しています。Squid はコンテナ上で動作しており、Squid へのアクセスは Giga1/1 を経由して行われます。Squid の IP アドレスは社内 LAN のクライアント PC と同一ネットワーク(192.168.1.0/24)のアドレス(192.168.1.254)を設定します。

クライアント PC が外部の Web サイトへアクセスする際には、プロキシである Squid へアクセスします。Squid はクライアント PC に変わって Web ページにアクセスし、Web サイトの情報をクライアントに返します。なお、Giga2/1 では NAT 変換され、装置から送信されるパケットの送信元アドレスは Giga2/1 につけられた global アドレス(192.0.2.2)になります。

(\*1) 機能説明書「2.19.3 ブリッジグループの装置内部構成」をご参照ください。

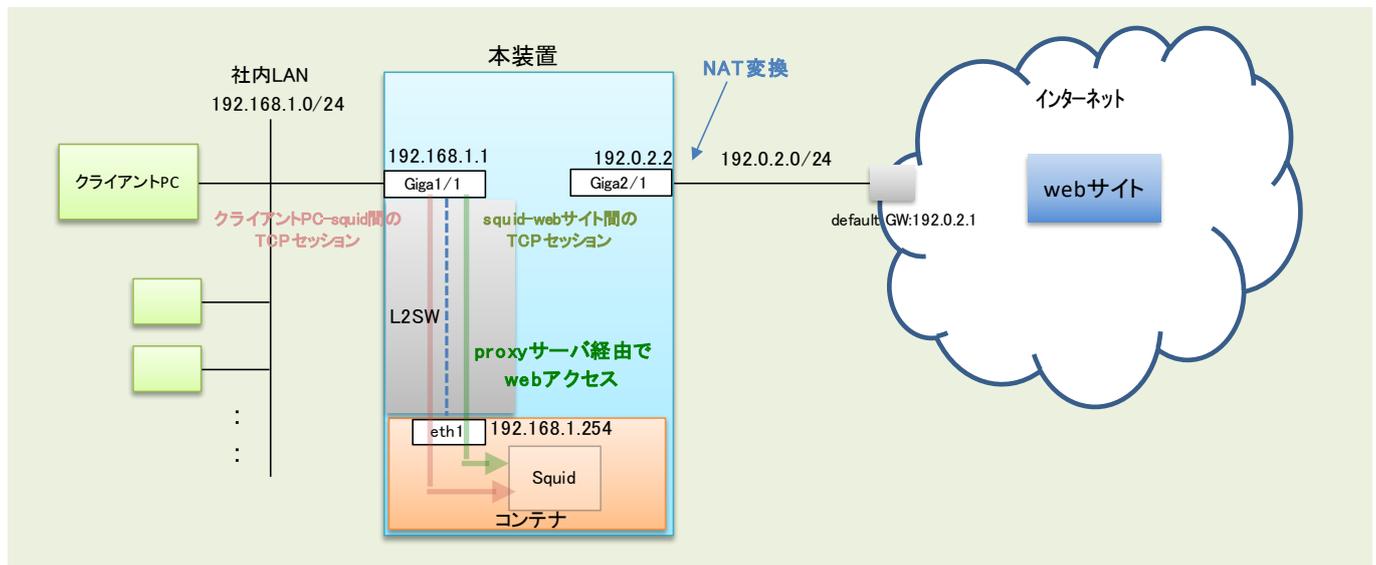


図 1. ネットワーク構成例

## 1. ルータ OS のインタフェース・コンテナ設定

コンテナ環境を使用できるようにルータ OS 側で、container enable や container-use のコンフィグ設定を行います。上記ネットワーク構成に合わせた FITELnet のコンフィグを以下に示します。

|   |  |
|---|--|
| container enable                        | ←コンテナのサービスを有効にする設定                           |
| !                                       |  |
| access-list 111 deny ip any any         | ←学習フィルタ設定                                    |
| access-list 121 spi ip any any          | ←学習フィルタ設定                                    |
| !                                       |  |
| ip route 0.0.0.0 0.0.0.0 100.1.1.1      | ←デフォルトルートをGWに設定                              |
| !                                       |  |
| ip dhcp server-profile DHCP_SERVER_PROF | ←DHCPサーバ機能のプロファイル                            |
| address 192.168.1.2 192.168.1.101       | ←DHCP払い出しアドレスレンジ                             |
| lease-time 3600                         | ←払い出し情報のリース時間(秒)                             |
| dns x.x.x.x                             | ←お客様に合わせてDNSサーバを指定<br>して下さい(x.x.x.xの部分)      |
| gateway 192.168.1.1                     |  |
| exit                                    |  |
| !                                       |  |
| ip nat list 1 192.168.1.0 0.0.0.255     | ←192.168.1.0/24をlist 1と定義                    |
| !                                       |  |
| logging buffer level informational      |  |
| !                                       |  |
| interface GigaEthernet 1/1              |  |
| vlan-id 1                               |  |
| bridge-group 1                          |  |
| channel-group 1                         |  |
| container-use                           | ←コンテナに論理インタフェースを設定                           |
| exit                                    |  |
| !                                       |  |
| interface GigaEthernet 2/1              |  |
| vlan-id 21                              |  |
| bridge-group 21                         |  |
| channel-group 21                        |  |
| ip access-group 111 in                  | ←WAN側からのアタックを防ぐための<br>学習フィルタ設定               |
| ip access-group 121 out                 |  |
| exit                                    |  |
| !                                       |  |
| interface Port-channel 1                |  |
| ip dhcp service server                  | ←Giga1/1にDHCPサーバ機能を設定                        |
| ip dhcp server-profile DHCP_SERVER_PROF | ←DHCPサーバ機能のプロファイル名を指定                        |
| ip address 192.168.1.1 255.255.255.0    |  |
| exit                                    |  |
| !                                       |  |
| interface Port-channel 21               |  |
| ip address 192.0.2.2 255.255.255.0      |  |
| ip nat inside source list 1 interface   | ←list 1に含まれる送信元アドレスを<br>インタフェースのアドレスにNAT変換する |
| exit                                    |  |
| !                                       |  |
| end                                     |  |

コンテナ環境では、"eth + bridge-group 番号"が名前となるインタフェースが生成されます（上記の例では、"eth1"が生成されます）。

詳細は、『コンテナ型仮想環境の使用法』の『1.2.ルータ OS の設定(CLI)』を参照ください。

## 2. コンテナ環境のネットワーク設定

下記のコンフィグを追加設定し反映することで、ルータ OS 側からコンテナ環境のネットワーク設定 (DNS、IP アドレス、ゲートウェイ) を行います。

|  |   |
|--|---|
| container configuration                | ←コンテナ設定モードへ移行                                     |
| dns x.x.x.x                            | ←コンテナに登録するDNSサーバアドレスを指定<br>お客様に合わせてDNSサーバを指定して下さい |
| !                                      | (x.x.x.xの部分)                                      |
| interface 1                            | ←Giga1/1とコンテナeth1が紐づけられる                          |
| bridge-group 1                         | ←IP addressを192.168.1.254に設定                      |
| ip address 192.168.1.254 255.255.255.0 | ←gatewayをGiga1/1(192.168.1.1)に設定                  |
| ip gateway 192.168.1.1                 |   |
| exit                                   |   |
| !                                      |   |
| exit                                   |   |
| !                                      |   |
| end                                    |   |

## 3. Squid の設定と起動

以下の手順で Squid の設定と起動を行います。

### 1) コンテナ環境へ移行

ルータ OS の CLI 上で『container attach』コマンドでコンテナ環境内のシェルを起動します。

### 2) Squid 待ち受けアドレス/ポート設定

コンテナ環境で/etc/squid フォルダ配下の設定ファイル (squid.conf) を編集し、”http\_port 3128”と記載されている箇所を、”http\_port 192.168.1.254:3128”に書き換えます。こうすることで、squid がクライアントからのリクエストを待ち受けるアドレスとポートが 192.168.1.254:3128 に設定されます。

(HTTP、HTTPS ともこのアドレス/ポート番号で待ち受け可能)

```

:
# And finally deny all other access to this proxy
http_access deny all

# Squid normally listens to port 3128
http_port 192.168.1.254:3128
# Uncomment and adjust the following to add a disk cache directory.
#cache_dir ufs /var/cache/squid 100 16 256
:

```

←コンテナアドレス (192.168.1.254) を追記

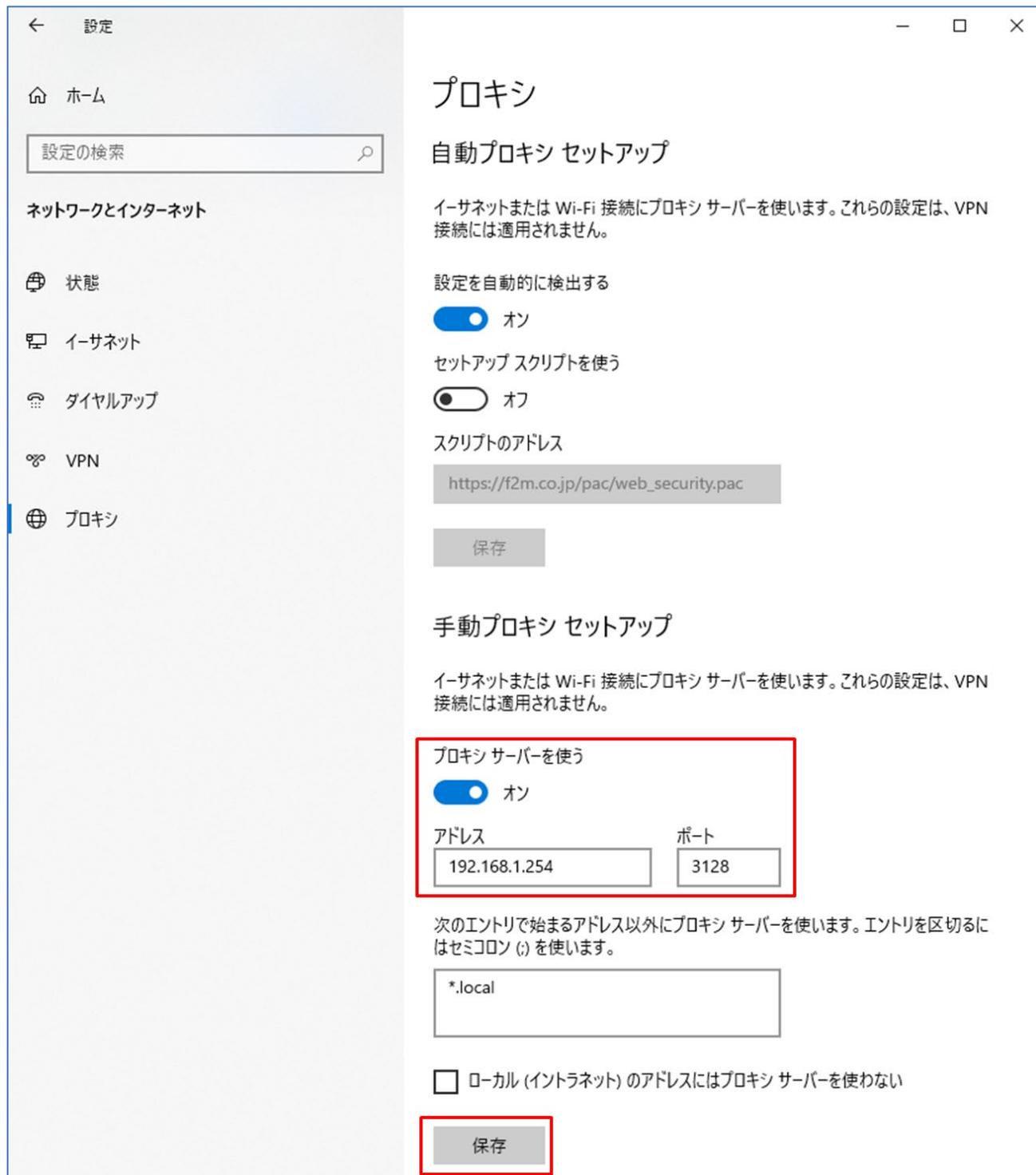
### 3) Squid 起動

『service squid start』コマンドで squid を起動します。

Squid は装置起動時は stop 状態ですので、装置再起動やコンテナ再起動した際は『service squid start』コマンドで都度起動が必要となります。

## 4. クライアント PC 設定

windows10 を例にとると、[スタート]メニューを右クリックし、[設定]アイコンをクリックします。設定ウインドウが開いたら、[ネットワークとインターネット]をクリックします。左側の[プロキシ]をクリックすると、下記のようにプロキシの設定画面が表示されます。プロキシサーバーを使うをオンにしてアドレス(192.168.1.254)とポート(3128)を設定して保存をクリックします。



上記設定、ブラウザから web サイトへアクセスして下さい。Squid を通してアクセスが可能になります。

## 5. Squid アクセスログ

コンテナ環境で/var/log/squid フォルダ配下の access.log がアクセスログになります。squid を通じてアクセスした端末、アクセスサイト等の情報が記録されています。

```
~ # cat /var/log/squid/access.log
1624520116.876 21 192.168.1.101 TCP_MISS/200 345 GET http://detectportal.firefox.com/success.txt? - HIER_DIRECT/34.107.221.82 text/plain
1624520117.370 1545 192.168.1.101 TCP_TUNNEL/200 5470 CONNECT mtalk.google.com:443 - HIER_DIRECT/108.177.97.188 -
1624520117.900 0 192.168.1.101 TCP_DENIED/403 3839 CONNECT mtalk.google.com:5228 - HIER_NONE/- text/html
1624520117.975 60 192.168.1.101 TCP_MISS/500 388 HEAD http://qzrppjbjqh/ - HIER_NONE/- text/html
1624520117.976 60 192.168.1.101 TCP_MISS/500 388 HEAD http://omdwemykabkk/ - HIER_NONE/- text/html
1624520117.977 61 192.168.1.101 TCP_MISS/500 388 HEAD http://ozxwfraoaabopx/ - HIER_NONE/- text/html
1624520118.432 37 192.168.1.101 TCP_MISS/500 388 HEAD http://kydatsnvet/ - HIER_NONE/- text/html
1624520118.436 40 192.168.1.101 TCP_MISS/500 388 HEAD http://toszxbjyji/ - HIER_NONE/- text/html
1624520118.439 42 192.168.1.101 TCP_MISS/500 388 HEAD http://fxknfsvfayz/ - HIER_NONE/- text/html
1624520118.929 863 192.168.1.101 TCP_TUNNEL/200 859 CONNECT fonts.gstatic.com:443 - HIER_DIRECT/216.58.220.131 -
1624520118.929 1028 192.168.1.101 TCP_TUNNEL/200 3375 CONNECT www.furukawa.co.jp:443 - HIER_DIRECT/18.176.183.51 -
1624520118.930 1029 192.168.1.101 TCP_TUNNEL/200 19224 CONNECT www.furukawa.co.jp:443 - HIER_DIRECT/18.176.183.51 -
1624520118.930 624 192.168.1.101 TCP_TUNNEL/200 1485 CONNECT www.google-analytics.com:443 - HIER_DIRECT/172.217.175.46 -
```

## 6. Squid に接続可能なクライアント

Squid は、デフォルトの設定で 192.168.0.0/16 が localnet という名称でアクセスリスト登録されており、“http\_access allow localnet”の行で localnet で登録された 192.168.0.0/16 を含むホストがアクセスすることを許可しています（下図①、④参照）。そのため 192.168.0.0/16 のネットワークのクライアントを接続する場合には特に設定の追加変更なくアクセスが可能です。また、http(ポート番号 80)や https(ポート番号 443)へのアクセスもデフォルトで可能となっています（下図②、③参照）。

許可するホストやポート番号を限定、または許可の範囲を広げたい場合は/etc/squid フォルダ配下の設定ファイル (squid.conf) を書き換えることで対応することができます。設定変更後は、『service squid restart』で設定を反映して下さい。

```
~ # cat /etc/squid/squid.conf
#
# Recommended minimum configuration:
#
# Example rule allowing access from your local networks.
# Adapt to list your (internal) IP networks from where browsing
# should be allowed
acl localnet src 0.0.0.1-0.255.255.255 # RFC 1122 "this" network (LAN)
acl localnet src 10.0.0.0/8 # RFC 1918 local private network (LAN)
acl localnet src 100.64.0.0/10 # RFC 6598 shared address space (CGN)
acl localnet src 169.254.0.0/16 # RFC 3927 link-local (directly plugged) machines
acl localnet src 172.16.0.0/12 # RFC 1918 local private network (LAN)
acl localnet src 192.168.0.0/16 # RFC 1918 local private network (LAN)
acl localnet src fc00::/7 # RFC 4193 local private network range
acl localnet src fe80::/10 # RFC 4291 link-local (directly plugged) machines

acl SSL_ports port 443
acl Safe_ports port 80 # http
acl Safe_ports port 21 # ftp
acl Safe_ports port 443 # https
acl Safe_ports port 70 # gopher
acl Safe_ports port 210 # wais
```

① 192.168.0.0/16 を localnet の名称でアクセスリスト登録

② 宛先ポートとして 80 (http) や 443 (https) を Safe\_ports の名称でアクセスリスト登録

```
acl Safe_ports port 1025-65535 # unregistered ports
acl Safe_ports port 280      # http-mgmt
acl Safe_ports port 488      # gss-http
acl Safe_ports port 591      # filemaker
acl Safe_ports port 777      # multiling http

#
# Recommended minimum Access Permission configuration:
#
# Deny requests to certain unsafe ports
http_access deny !Safe_ports ← ③Safe_ports 登録以外のポートアクセスを制限

# Deny CONNECT to other than secure SSL ports
http_access deny CONNECT !SSL_ports

# Only allow cachemgr access from localhost
http_access allow localhost manager
http_access deny manager

# We strongly recommend the following be uncommented to protect innocent
# web applications running on the proxy server who think the only
# one who can access services on "localhost" is a local user
#http_access deny to_localhost

#
# INSERT YOUR OWN RULE(S) HERE TO ALLOW ACCESS FROM YOUR CLIENTS
#

# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
# from where browsing should be allowed
http_access allow localnet ← ④localnet 登録ホストのアクセスを許可
http_access allow localhost

# And finally deny all other access to this proxy
http_access deny all

# Squid normally listens to port 3128
http_port 192.168.1.254:3128

# Uncomment and adjust the following to add a disk cache directory.
#cache_dir ufs /var/cache/squid 100 16 256

# Leave coredumps in the first cache dir
coredump_dir /var/cache/squid
```

```
#
# Add any of your own refresh_pattern entries above these.
#
refresh_pattern ^ftp:          1440  20%  10080
refresh_pattern ^gopher:      1440  0%   1440
refresh_pattern -i (/cgi-bin/|?) 0    0%   0
refresh_pattern .              0     20%  4320
~ #
```

## 7. Squid ベーシック認証

下記の設定を行うことで、クライアント PC からのアクセスに対し Squid でベーシック認証を行うこともできます。

### 1) squid.conf にベーシック認証設定を追加

コンテナ環境で/etc/squid フォルダ配下の設定ファイル (squid.conf) を編集し、” http\_access allow localnet”と記載されている行の前に下記設定を追加します。

```
:
# Example rule allowing access from your local networks.
# Adapt localnet in the ACL section to list your (internal) IP networks
# from where browsing should be allowed
auth_param basic program /usr/lib/squid/basic_ncsa_auth /etc/squid/passwords
acl authenticated proxy_auth REQUIRED
http_access allow authenticated
http_access allow localnet
http_access allow localhost
:
```

←設定追加

### 2) ベーシック認証用のアカウント/パスワードを設定

コンテナ環境で以下のコマンドを実行し、ベーシック認証用のアカウントとパスワードを設定します。(例ではアカウントを”test”としパスワードを”secret”にしています。)

```
~ # htpasswd -c /etc/squid/passwords test          ←testというアカウントを作成
New password:                                     ←secret と入力
Re-type new password:                             ←もう1度 secret と入力
Adding password for user test
~ #
```

“test”というアカウントが作成されているか下記コマンドで passwords ファイルの中身を確認します。

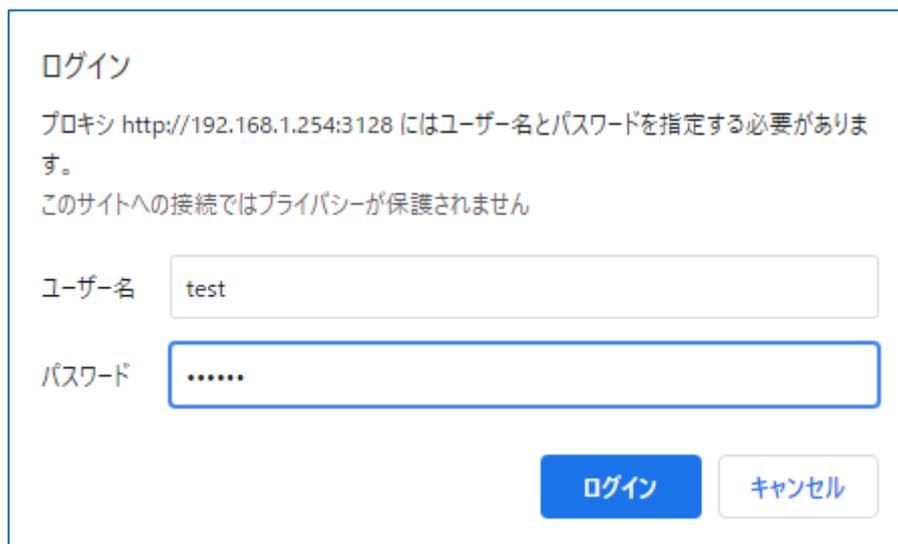
```
~ # cat /etc/squid/passwords                      ←アカウントが設定されたか確認
test:$apr1$B.k0HWcf$lrwNJyGZiZFYOpjxxBVAz0     ←testというアカウントが作成されている
~ #
```

### 3) squid 再起動

『service squid restart』コマンドで squid を再起動します。

4) ベーシック認証ログイン

ブラウザ (例では chrome) を開いて適当な URL へアクセス(例えば `www.furukawa.co.jp` すると、下記のようなログイン画面が表示されるので、ユーザ名とパスワードのウインドウに先ほど設定した”test”と”secret”を入力してログインを押すと、web アクセスが可能になります。



The screenshot shows a login dialog box with the following content:

- Title: ログイン
- Message: プロキシ http://192.168.1.254:3128 にはユーザ名とパスワードを指定する必要があります。  
このサイトへの接続ではプライバシーが保護されません
- User Name field: ユーザー名 test
- Password field: パスワード .....
- Buttons: ログイン (Login) and キャンセル (Cancel)

以上