

# example\_notebook\_1

July 27, 2023

## 1 Examples for using the wgconfig library for reading data

We have a WireGuard example configuration in “wgtest.conf” with an interface section, two regular peers, one disabled peer and a lot of comments. It has the following content:

```
[5]: from pprint import pprint
with open('./wgtest.conf', 'r') as f:
    print(f.read())

# This is a first comment
[Interface]
# This is a second comment
PrivateKey = 6FYKQKEtGFAB5HSwyj5cQl3wgS1E9d6SqVjdVksOn2s=
# PublicKey = S/aHw6LOM+yq5m9qikcfy++dhPdw7tHuNMPgwQkEdSo=
ListenPort = 51820
Address = 192.0.2.1/24 # end-of-line comment
Address = fe80::1/64

# This is a third comment
[Peer]
Endpoint = 192.168.0.2:51820
# PrivateKey = cKqe3xDFsKlMwlQfVJAnbNhiGFV57FnfLykiBtrnumY=
PublicKey = XWItB4SR1qwGbGn59oRE6TBlTYHQF0pDy1x63dlr5nA=
AllowedIPs = fe80::2/128, 9999::2/128
PersistentKeepalive = 25
# This is a forth comment

[Peer]
# This is a fifth comment
Endpoint = 192.168.0.3:51820
# PrivateKey = iJQkwzeB2+/lGyGPTM23Wes5Kg0n+LgXMqK8XAwWt14=
PublicKey = eBvBVLo6wH0XkBfIjeLPf8ydBTfU/gMqJOH4nmVXcDE=
AllowedIPs = fe80::3/128
AllowedIPs = 9999::3/128
PersistentKeepalive = 25

#! [Peer]
#! Endpoint = 192.168.0.4:51820
```

```

#! # PrivateKey = iAgWkT6/Fn0+kcNcD65SKpjcAweLmcppVE4IEHxa73o=
#! PublicKey = ivBD0+pT2m4W5b17ApNaC3BybEtYa1fvNpA4h+tHyy8=
#! AllowedIPs = fe80::4/128
#! AllowedIPs = 9999::4/128
#! PersistentKeepalive = 25

```

### 1.0.1 Let's import the wgconfig library and read that file

```

[6]: import wgconfig
wc = wgconfig.WGConfig('./wgtest.conf')
wc.read_file()

```

### 1.0.2 Interface data parsed by wgconfig

```

[7]: pprint(wc.get_interface())

{'Address': ['192.0.2.1/24', 'fe80::1/64'],
 'ListenPort': 51820,
 'PrivateKey': '6FYKQKEtGFAB5HSwyj5cQl3wgS1E9d6SqVjdVks0n2s='}

```

### 1.0.3 Get a list of all peers - just active ones or also including disabled ones

```

[8]: pprint(wc.get_peers())
print()
pprint(wc.get_peers(include_disabled=True))

['XWItB4SR1qwGbGn59oRE6TB1TYHQF0pDy1x63dlr5nA=',
 'eBvBVLo6wH0XkBfIjeLPf8ydBTfU/gMqJOH4nmVXcDE=']

['XWItB4SR1qwGbGn59oRE6TB1TYHQF0pDy1x63dlr5nA=',
 'eBvBVLo6wH0XkBfIjeLPf8ydBTfU/gMqJOH4nmVXcDE=',
 'ivBD0+pT2m4W5b17ApNaC3BybEtYa1fvNpA4h+tHyy8=']

```

### 1.0.4 Get the data of a single peer

```

[9]: pprint(wc.get_peer('XWItB4SR1qwGbGn59oRE6TB1TYHQF0pDy1x63dlr5nA='))

{'AllowedIPs': ['fe80::2/128', '9999::2/128'],
 'Endpoint': '192.168.0.2:51820',
 'PersistentKeepalive': 25,
 'PublicKey': 'XWItB4SR1qwGbGn59oRE6TB1TYHQF0pDy1x63dlr5nA='}

```

Note that attributes with the same name occurring on different lines are returned as list (see the “AllowedIPs” attribute lines of first peer).

```

[10]: pprint(wc.get_peer('eBvBVLo6wH0XkBfIjeLPf8ydBTfU/gMqJOH4nmVXcDE='))

{'AllowedIPs': ['fe80::3/128', '9999::3/128'],
 'Endpoint': '192.168.0.3:51820',

```

```
'PersistentKeepalive': 25,  
'PublicKey': 'eBvBVLo6wH0XkBfIjeLPf8ydBTfU/gMqJOH4nmVXcDE='}
```

Note that comma-separated values are also returned as list (see the “AllowedIPs” attribute of second peer).

### 1.0.5 Get data of all peers

```
[11]: pprint(wc.get_peers(keys_only=False))
```

```
{'XWItB4SR1qwGbGn59oRE6TB1TYHQF0pDy1x63dlr5nA=': {'AllowedIPs': ['fe80::2/128',  
                                                                    '9999::2/128'],  
                                                       'Endpoint':  
                '192.168.0.2:51820',  
                                                       'PersistentKeepalive': 25,  
                                                       'PublicKey':  
                'XWItB4SR1qwGbGn59oRE6TB1TYHQF0pDy1x63dlr5nA='},  
 {'eBvBVLo6wH0XkBfIjeLPf8ydBTfU/gMqJOH4nmVXcDE=': {'AllowedIPs': ['fe80::3/128',  
                                                                    '9999::3/128'],  
                                                       'Endpoint':  
                '192.168.0.3:51820',  
                                                       'PersistentKeepalive': 25,  
                                                       'PublicKey':  
                'eBvBVLo6wH0XkBfIjeLPf8ydBTfU/gMqJOH4nmVXcDE='}}}
```

```
[ ]:
```