

Public Git User's Manual

Table of Contents

- 1 [Overview](#)
- 2 [Software Requirements](#)
- 3 [Accounts and Key](#)
- 4 [Start with Gitaccess](#)
- 5 [Repo the Code](#)
- 6 [Q&A](#)

Overview

Our previous release method is releasing a Tarball of patches based on Google's code. This method makes customer difficult to restore the code base. Different kind of issues can be met. GIT is a fast distributed revision control system. Using public release GIT, we will release the source code though GIT server. Customer can use the repo command to download the code from our server which is same as they download from Google's server.

This manual is designed to help customer how to download the code from our Public GIT server.

1 Software Requirements

1.1 Host PC with operating system – Ubuntu

1.2 Marvell recommends using Ubuntu 10.04 and manually installing the Git, Repo and Python. Follow the instructions on <http://source.android.com/source/download.html> to set up the Android build environment on Ubuntu Linux.

2 Accounts and Key

2.1 En extranet account on <https://extranet.marvell.com/>, it's an authorized site for our Customers, Representatives and Distributors who are under a Marvell Non-Disclosure Agreement.

2.2 A DSA or RSA Key for SSH authentication.

First, you will need to generate the local DSA or RSA key

```
#ssh-keygen -t dsa
```

```
#ssh-keygen -t rsa
```

*the keys should be generated at ~/.ssh/id_dsa.pub and ~/.ssh/id_rsa.pub} the application would prompt you to enter a passphrase, please **DO NOT** enter any password.*

For example:

```
XXX@XXX-desktop:~$ ssh-keygen -t rsa
```

```
Generating public/private dsa key pair.
```

```
Enter file in which to save the key (/home/XXX/.ssh/id_rsa):
```

```
Enter passphrase (empty for no passphrase):
```

```
Enter same passphrase again:
```

```
Your identification has been saved in /home/XXX/.ssh/id_rsa.
```

```
Your public key has been saved in /home/XXX/.ssh/id_rsa.pub.
```

```
The key fingerprint is:
```

```
96:f6:4a:10:9f:98:5e:e0:1d:f9:a1:d9:c8:bb:64:e7 XXX@XXX-desktop
```

```
The key's randomart image is:
```

```

+--[ DSA 1024]-----+
|      |
|      .      |
|    o o .    |
|  . O X .    |
|    = S o    |
|  . = o      |
|    . = o    |
|      + =    |
|    o E      |
+-----+
    
```

3 Start with the gitaccess

3.1 Login the Gitaccess on <http://github.marvell.com> by extranet account.

To access the <http://github.marvell.com>, it requires IP verification by **MARVELL** service desk.

Please contact the FAE to submit a request. To find out your own IP address, you can get it by access: <http://whatismyipaddress.com/>



The image shows a login form titled "GIT Access Login". It has two input fields: "User ID" and "Password". The "User ID" field contains the text "Extranet User ID" and is highlighted with a red border. The "Password" field contains the text "Extranet PW" and is also highlighted with a red border. Below these fields is a "Log In" button with a right-pointing arrow.

3.2 First time login the gitaccess, it requires to create a user name and upload your ssh Public Key.

To create a DSA Key for SSH authentication. Please check [2.2](#)

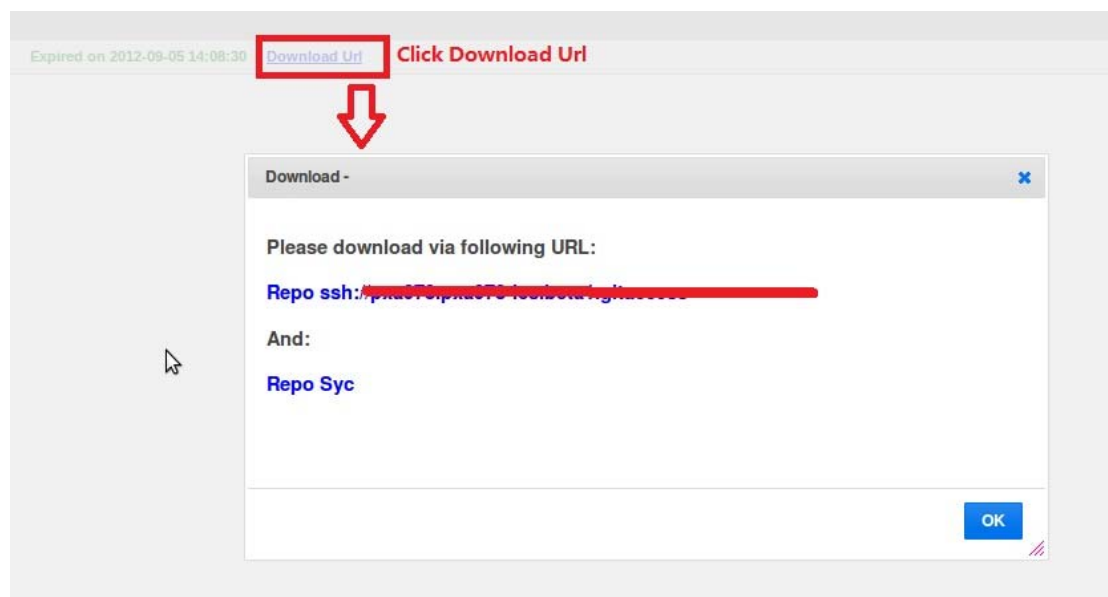
A web form titled 'Gerrit Access Account Register' in red. Below the title, it says 'Please register with your Gerrit user name, full name and public rsa key:'. There are two input fields: 'User Name:' and 'Public Key:'. The 'User Name:' field is highlighted with a red rectangle. To the right of the 'Public Key:' field is a 'Browse...' button. At the bottom left is a 'Submit' button with a red arrow icon. To the right of the 'Submit' button is a link: 'How to generate a rsa key file?'.

Note: by default, the public key id_dsa.pub is in the hidden folder ~/.ssh, so please copy it to another location, then you can find and choose the pub key from the button 'Browse...'

3.3 Read the EULA and sign up (by Click “**Agree**”) before downloading the code. This is same as the Click through License used on extranet.

The authentication will be expired after 48 hours.

3.4 Get URL to download the code after signing up the EULA



4 Repo the Code

4.1 Use the URL posted to download the manifest.

It's able to test the accessibility for "github.marvell.com" by: "ssh {username}@github.marvell.com -p 29418" before downloading the manifest and tree.

Run:

```
$repo init -u ssh://XXX@github.marvell.com:29418/platform/manifest.git -b  
XXX.XXX
```

4.2 Getting the files. To pull down files to your working directory from the repositories as specified in the default manifest

Run:

```
$repo sync
```

5 Q&A

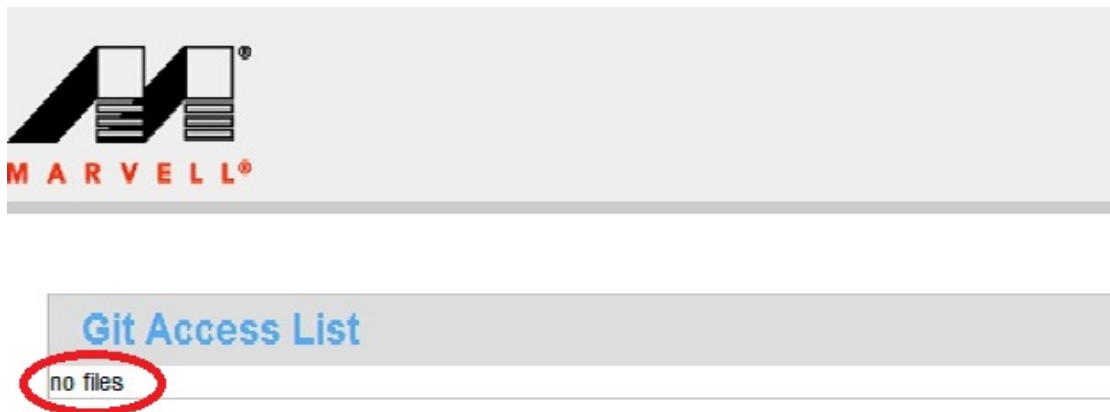
Q: Why I cannot access the github.marvell.com?

A: Please check [3.1](#)

Q: I can access the github.marvell.com, but I can't repo the code?

A: Please check [3.3](#) and [3.4](#) to make sure the EULA has been accepted and not expired.

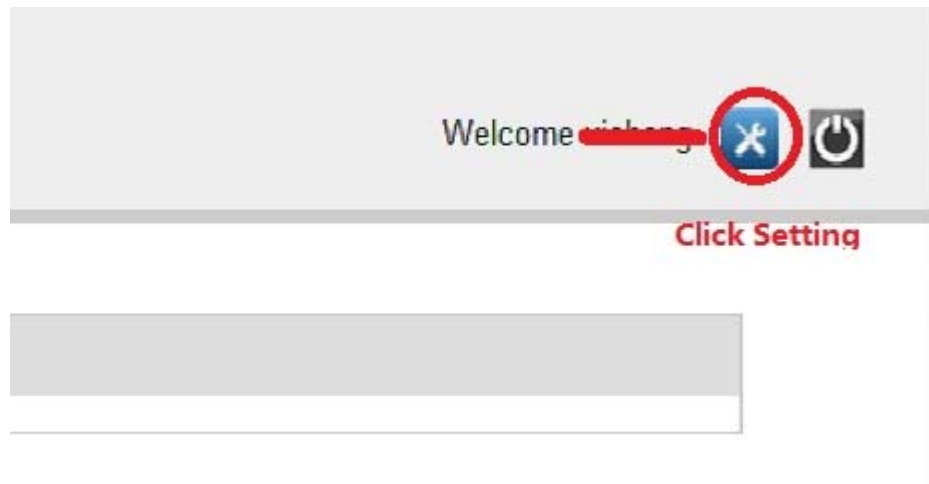
Q: Why it shows no files after login in the gitaccess by extranet account? Such as:



A: Please contact the FAE to add your account into directly project group.

Q: Is that able to download the code from flexible Host PC by one account?

A: Yes. Please upload the multiple Keys by clicking setting menu in the gitaccess side.



Q: How to optimize the download time if my host got narrow bandwidth to github.marvell.com?

A: It's able to minimize the download size from **MARVELL** by:

repo the code from google to ~/mirror/

- a) `mkdir -p ~/mirror`
- b) `cd ~/mirror`
- c) `repo init -u https://android.googlesource.com/mirror/manifest --mirror`
- d) `repo sync`

repo the code from github.marvell.com and using --reference to refer to ~/mirror.

- e) `repo init -u ssh://XXX@github.marvell.com:29418/platform/manifest --reference ~/mirror`
- f) `repo sync`