Urchin Software from Google

Install Guide for FreeBSD/Linux

Software Version : 7.0
Revision : June 15th, 2010

Important Note for Urchin 6.500 Users

The Urchin 6.501 release fixed several issues in the CPC scheduler introduced in the Urchin 6.5.0 version. If you are upgrading from Urchin 6.5 to Urchin 7, your already-processed CPC data might be incorrect or incomplete, resulting in erroneous results in the Urchin reports. In order to get correct data from Google Adwords, we recommend that you either (a) install Urchin 7 in a new location or (b) if you wish to install on an existing 6.5 instance, delete the old CPC data. In this latter case, delete all contents of the following directories:

```
<path_to_urchin_6500_installation>/data/cpc/...
<path_to_urchin_6500_installation>/data/history/cpc/...
```

No action is required if you are not upgrading from Urchin 6.5 (i.e. Urchin 6.501 and later releases OR Urchin 6.402 and prior releases).

Important Note for FreeBSD Users

1) FreeBSD has a hard-coded process datasize limit of 500MB, which is too small to allow Urchin to run due to the size of the geodata that has to be memory resident during log processing. The error you will see during runtime is:

```
ERROR: (8011-323-1057) Failed to allocate memory
```

To address this, you'll need to increase the FreeBSD system's default process datasize, which you can do by editing the /boot/loader.conf file and adding these lines:

```
# Increase max process data segment size to 1GB
kern.maxdsiz="1073741824"
```

You'll need to reboot afterwards.

2) FreeBSD has special driver called *accf_http*. It will buffer incoming connections until a certain number of complete HTTP requests arrive to speed up and optimize performance. By default this driver is not loaded to the kernel and you will see the following warning:

```
[warn] (2)No such file or directory: Failed to enable the
'httpready' Accept Filter
```

To avoid this, you'll need to enable the HTTP Accept Filter FreeBSD kernel module (*accf_http*), which you can do by editing the /boot/loader.conf file and adding these lines:

```
# Wait for full HTTP request accept filter
accf_http_load="YES"
```

You'll need to reboot afterwards.

WARNING!!! Be very careful when changing system limits like this. If you set kern.maxdsiz to be too large, your system may become unbootable. We **strongly recommend** that you read the FreeBSD documentation before making this modification, and assess the potential risks as they apply to your site-specific configuration.

Overview of Installation Steps

- Install the desired database server on the Urchin system or add Urchin's database to an existing server. Currently supported database engines:
    ◦ MySQL 5.0.3 or later
    (**Note**: Beginning with Urchin 6.6, MySQL 4.x.x is no longer supported)
    ◦ PostgreSQL 8.2.5 or later
- Create the Urchin database and user in on the database server and grant access permissions
- Unpack the Urchin distribution in a temporary directory (**Note**: Urchin 7 64-bit installers don't have platform checking routines. So, check the operating system type before running the appropriate installer package.)
- Run the install.sh script
- License the Urchin installation

Option 1: Using existing MySQL database with Urchin

- If you don't already have a MySQL server set up on your system, you can obtain a distribution from http://dev.mysql.com/downloads/
- Install the MySQL server per the instructions on the MySQL site
- Make sure that you know the MySQL database administrator login (e.g. root) and password
- Choose the database name, local MySQL user and password you will use with Urchin. Save this information, as you'll need it later during the Urchin installation. The example below creates a database named **urchin** and a MySQL user name **urchin** with password **urchinpassword** (but you should obviously choose your own password!)
- Create the Urchin database, user and permissions with the following commands:

```
root@yourserver# mysql -u root -p
mysql> create database urchin character set utf8;
mysql> GRANT ALL ON urchin.* to 'urchin'@'localhost' IDENTIFIED BY
'urchinpassword';
```

- Determine the pathname of the MySQL socket for communicating with your MySQL server. You can usually find this in the my.cnf file for your MySQL installation, look for the line that looks like socket=/var/lib/mysql/mysql.sock

Option 2: Using existing PostgreSQL database with Urchin

- If you don't already have a PostgreSQL server set up on your system, you can obtain a distribution from http://www.postgresql.org.
- Install the PostgreSQL server per the instructions on the PostgreSQL site

- Make sure that the psql command is in your path and you know the PostgreSQL database administrator login (e.g. pgsql) and password
- Choose the database name, local PostgreSQL user and password you will use with Urchin. Save this information, as you'll need it later during the Urchin installation. The example below creates a database named **urchin** and a PostgreSQL user name **urchin** with password **urchinpass** (but you should obviously choose your own password!)
- Create the Urchin database, user and permissions with the following commands:

```
root@yourserver# psql -U pgsql -d postgres
postgres=# CREATE DATABASE urchin WITH ENCODING 'UTF8';
postgres=# CREATE USER urchin WITH PASSWORD 'urchinpass';
postgres=# GRANT ALL ON DATABASE urchin TO urchin;
postgres=# \connect urchin;
postgres=# CREATE LANGUAGE 'PLPGSQL';
```

Option 3: Using bundled PostgreSQL server with Urchin

- Urchin 7 installer comes with bundled PostgreSQL server as an option during Urchin installation.
- In order to use this option Urchin installer should be invoked under user with root privileges. Installer will prompt you to provide port number and password for the user "postgres". **Note:** If the user "postgres" already exists in the system, then existing password needs to be provided. If user "postgres" doesn't exist in the system, it will be created with provided password.
- The following limitations apply to the bundled PostgreSQL server:
  - It can't be installed on FreeBSD.
  - It requires glibc version 2.3.3 or above.
  - It should not be installed in upgrade mode.
  - Root privileges are required for installation.

Installing Urchin

- Make sure that you have the Urchin database name, user and password that you selected above
- Create a temporary directory and unpack the Urchin distribution into it:

```
mkdir /var/tmp/urchintemp
cd /var/tmp/urchintemp
tar xzvf urchin7000_<platform>_installer.tar.gz
```

- **Important!** To ensure correct installation, make sure that there is no 'active' instance of Urchin 7 on the target machine. Disable any existing version of Urchin 7 with the following command:

```
/path/to/urchin7/bin/urchinctl stop
```

- If you need to run multiple instances of Urchin 7 on the same machine, then the instances should (1) be installed in different locations, (2) have different web server ports and (3) connect to different databases.
- Urchin copies all necessary files to the target folders and overwrites any existing files. In order to keep any existing customization intact, you'll need to install Urchin 7 in a new directory, or rename the existing installation directory.

- Urchin 7 can either be installed from scratch or upgraded from an existing version of Urchin 6.

New Installation

- During 'New' installation, Urchin prompts you for all configuration parameters, such as web server port, db configuration settings, etc.
- If you are installing bundled PostgreSQL server, then make sure that there is no existing PostgreSQL server on the system. To install, select an option "Install PostgreSQL" and provide PostgreSQL server details such as port number and administrator password. **Note:** Please refer to http://www.postgresql.org if you have any problems installing the bundled PostgreSQL server.
- During 'New' installation, Urchin prompts you to "Initialize database tables". If you want to keep an existing DB configuration, then select 'No'. In this case, Urchin preserves the existing configuration (by not recreating the existing tables), but it may need to create 'additional tables' or 'columns in existing tables'.
- Now run the Urchin installer. You can either run it interactively and follow the prompts by typing:

```
./install.sh
```

or you can perform a complete install directly from the command line, supplying all the necessary parameters for the install. You can obtain a list of the command line parameters by typing:

```
./install.sh -h
```

- Once the installer completes, you'll have a running version of Urchin on your system. To finish the setup, point your favorite web browser at this URL:

```
http://yourserver:<port>
```

where <port> is the network port you chose during the installation (default is 9999).
- If you wish to restore your profile reporting data from an older version of Urchin 7, copy the history, reports and any existing cpc directories from /path/to/old-urchin7/data/ to your new /path/to/new-urchin7/data/ directory.
- Click on the **Obtain Demo License** link and follow the licensing steps to enable Urchin and perform the initial setup steps.

Upgrading & Upgrade Limitations

- If you are upgrading Urchin 6.x to Urchin 7, note that the Urchin 7 installer can only be used for upgrading certain versions of Urchin 6. The following versions can be upgraded:
  - Urchin 6.6 - Service Pack 3   : Version 6.603
  - Urchin 6.6 - Service Pack 2   : Version 6.602
  - Urchin 6.6 - Service Pack 1   : Version 6.601
  - Urchin 6.6 - English Release   : Version 6.600
  - Urchin 6.5 - Service Pack 2   : Version 6.501
  - Urchin 6.5   : Version 6.500 (see important note above)

- Urchin 6 - Service Pack 1   : Version 6.402
- Urchin 6 - International Release : Version 6.401
- Urchin 6 - English Release   : Version 6.400
- To begin Urchin 7 upgrade installation (**Note:** Stopping/uninstalling the existing Urchin 6 instance is not required):
  - Run the Urchin 7 installer

```
./install.sh
```

  - Select language for Urchin installer.
  - Select an upgrade option and provide location for existing Urchin 6 instance. **Note:** Urchin automatically determines database configuration and stops Urchin 6 instance.
  - Follow the interactive prompts and perform an upgrade. **Note:** The Urchin 7 installer installs the new Urchin 7 files and reuses the configuration of the existing Urchin 6 instance.
  - Log into Urchin 7 as super admin user and configure a new Urchin 7 license. **Note:** Urchin 6 licenses are not valid for Urchin 7. In order to upgrade regular Urchin 6 licenses to Urchin 7, please contact Urchin Support.
  - Start all Urchin 7 schedulers.

```
./urchinctl -s start
./urchinctl -c start
```

Configuring Tracking

Once Urchin is installed, there may be additional steps required to set up tracking on your website depending on your use case. See the Urchin Help Center's information on the UTM tracking method.

Case 1: Using Urchin with IP+User-Agent Tracking

No additional configuration is necessary. When adding profiles to Urchin, be sure to specify IP+User-Agent as the Visitor Tracking Method.

Case 2: Using Urchin with UTM Tracking

To use Urchin with UTM tracking, it is necessary to install a small piece of javascript tracking code on each of your website's pages (essentially the same method used by Google Analytics). See section "UTM Quick-Install (Apache)" for more details.

Case 3: Using Urchin concurrently with Google Analytics

Please note that the new tracking features in the Google Analytics "ga.js" tracking code are not backward compatible with Urchin 7. If you wish to track your website with both Google Analytics and Urchin Software, we recommend using the urchin.js-based tracking code for best results. If you choose to use ga.js, you will not be able to use new ga.js tracking features.
To use Urchin with a website that is already being tracked with Google Analytics, you just need to make a slight modification to the existing tracking code for Google Analytics. Here are the steps:
- Copy only the __utm.gif file from the util/utm directory of the Urchin distribution to the document root of your website (e.g. the top level directory of your content).

- Add the line _userv=2; as shown below to the existing tracking code on your website pages. Be sure not to modify any other parts of the existing tracking code!

```
<script src="http://www.google-analytics.com/urchin.js"
type="text/javascript">
<script type="text/javascript">
 _userv=2;
 _uacct="UA–XXXXXXX–Y";
 urchinTracker();
</script>
```

- When adding profiles to Urchin, be sure to specify Urchin Traffic Monitor (UTM) as the Visitor Tracking Method.

Note: For UTM-based tracking, Urchin works with both locally hosted urchin.js as well as with the one hosted by Google. It is recommend to use local urchin.js with `_userv=0`.

UTM Quick-Install (Apache)

The following is intended as a quick run-through on installing the **UTM** for websites running on an Apache server on all platforms.

Step 1: Copy UTM files to website document root : The files, urchin.js and __utm.gif are located in the "util/**utm**" directory in the Urchin distribution. Copy these two files to the main directory of your website content.

Step 2: Reference UTM in your HTML : Enter the following line in all of your HTML pages. While it can go anywhere in the pages, we recommend putting it in the <head> section. If you use a common include or template, you can enter it there. **IMPORTANT**: the GIF filename begins with two underscores.

```
<script src="/urchin.js" type="text/javascript"></script>
<script type="text/javascript">
                urchinTracker();
</script>
```

If you are using a package like "HTML Tidy", you may want to include the Javascript line in the HEAD area of your page, for instance:

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html;
charset=ISO-8859-1">
  <script src="/urchin.js" type="text/javascript"></script>
  <script type="text/javascript">
                urchinTracker();
  </script>
  ...
</head>
```

Step 3: Enable cookies in your Apache logging : If not already enabled, you can use the following httpd.conf example to enable cookie logging:

```
LogFormat "%h %v %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-
Agent}i\" \"%{Cookie}i\"" urchin
...
<VirtualHost 63.212.171.6>
ServerName www.mysite.com
DocumentRoot /www/www.mysite.com
CustomLog /var/logs/mysite.log urchin
</VirtualHost>
```

Step 4: Set Urchin Profile to use UTM : In the Urchin Administration interface, edit the profile in question. Set the Visitor Tracking Method to UTM. Set the UTM Domain to the address of your website without the www. When done click the Update button. That's it! Your website will now begin logging **UTM** data into your normal log file which will be identified the next time you run Urchin.

Is it working?
To see if the UTM is successfully making entries to your log file, examine the log after you have installed the UTM and clicked on a few pages of the site. You should see an entry similar to the following at the end of the log file:

```
... "GET /__utm.gif?..." 200 ..."__utma=..."
```

If you don't see the utma entries, be sure to check that cookies was enabled in the logging properly. If the status code is not 200 then check to make sure the files were properly copied to your document root.