



Basic PostgreSQL Administration

Frank Wiles

frank@revsys.com

Slides available at:

<http://www.revsys.com/talks/kulua/basic-postgresql/>

PostgreSQL In a Nutshell

- Full featured RDBMS
- Views, triggers, foreign keys, complex queries, transactions, MVCC
- Stored Procedures in several languages (perl, python, ruby, tcl, java, php, scheme, R, lolcode, etc.)
- Various replication methods (pgpool-II, Slony-II, Skytools/londiste)

Setting up a new database cluster

```
# initdb -D /path/to/database/data
```

Setting up a new database

```
# createdb <name>
```

Creating users

```
# createuser <username>
```

Restricting user access with pg_hba.conf

Type	Database	User	CIDR	Method
local	all	all		trust
host	foo	bar	10.0.0.1/32	md5
host	dev	all	10.0.0.0/24	trust

Restricting user actions

GRANT ALL ON TABLE accounting TO frank;

**GRANT SELECT ON TABLE accounting TO
PUBLIC;**

GRANT INSERT ON TABLE logs TO cronuser;

Performance Tuning

Config Option	Description
shared_buffers	Set to 10-20% of RAM
effective_cache_size	Set to ~75% of RAM
random_page_cost	Default is 4.0, work down to 2.0
work_mem	Set to amount of data to be sorted

Performance Tuning

For more information see:

<http://www.revsys.com/writings/postgresql-performance.html>

Simple Backups

```
# pg_dump <database> > backup.sql
```

```
# pg_dump <database> | gzip > backup.sql.gz
```

```
# pg_dumpall | gzip > fullbackup.sql.gz
```

Filesystem backups

1. Run `SELECT pg_start_backup('label');`
2. `tar -cvzf backup.tar.gz /path/to/data-directory`
3. Run `SELECT pg_stop_backup();`
4. Ensure you have the last file needed in the `pg_xlog/` directory

Warm Standby

Warm Standby

1. Master server generates WAL files as it normally would, but instead of deleting them when they are no longer useful to the master they are archived
2. Standby server pulls these archived log files and incorporates them as if the system had crashed
3. This process continues until the standby is needed

Setting up warm standby

1. Set `archive_mode=on`
2. Set `archive_command="rsync -arv %p /backup-location/%f </dev/null"`
3. Make a filesystem level backup from previous slide and transfer to standby server
4. Setup a `rsync` cron job on standby to pull these archived WAL files to recovery area and remove them from server with `--delete`

Setting up warm standby

5. Make a file named `recovery.conf` with the following configuration in it:

```
restore_command = '/usr/local/bin/pg_standby -d -s 2 -t /  
pgdata/trigger-end-recovery.5432 /pg_wal_archive %f %p %r  
2>>/pgdata/standby.log'
```

6. To end “recovery mode” and turn your standby into a live server simply touch the trigger file, in this case `/pgdata/trigger-end-recovery.5432`

Questions?