DALIBO
L'Expertise PostgreSQL

pgShark - Lighning talk

Ce document est téléchargeable gratuitement dans la base de connaissance DALIBO.
Pour toute information complémentaire, contactez : formation@dalibo.com

Table des matières

pgShark - Lightning talk

1 Presentation

Me: Jehan-Guillaume (ioguix) de Rorthais

Activities:

- **phpPgAdmin** dev & admin
- PgSQL extension to control pgPool: **pgpool_adm**
- full-time PostgreSQL lover at Dalibo

IRC: ioguix

Contact: jgdr@dalibo.com

2 pgShark?

"Extract and mess with PostgreSQL messages from network dump"

Initially based on PDML output from tshark (Wireshark project)

⇒ tshark + PgSQL == pgShark

3 Why?

Original goal:

- upgrade from PgSQL 8.1 ⇒ 8.4
- lotta bad implicit conversions all over the place
- devs need help to find those queries
- high loaded production server
  ⇒ can't afford logging all queries

Solution: extract SQL from network dump and test them.

4 Nowadays

New goals:

- generic PostgreSQL messages dissector
- mess with everything, not only queries
- having fun

5 History (1/2)

Ancestor (early 2010):

- initially a PHP script parsing PDML output from tshark
- Pros: quick dirty PHP hack
- Cons:
  - very slow
  - hard to extend
  - doesn't cover the whole protocol
  - PHP

6 History (2/2)

Current (early 2011):

- Perl + libpcap
- fast, complete, portable
- from your perl scripts:

```
use pgShark::Core
```

7 How? (1/3)

```
my $shark = pgShark::Core->new({
  'procs' => {
    'Query' => \&Query,
    'Execute' => \&Execute,
  },
  'host' => $args{'host'}, 'port' => $args{'port'}
});

sub Query {
  my $pg_msg = shift;
  printf "QUERY query=%s\n", $pg_msg->{'query'};
}

sub Execute {
  my $pg_msg = shift;
  printf "EXECUTE name='%s', nb_rows=%d\n",
    $pg_msg->{'name'}, $pg_msg->{'nb_rows'};
}
```

8 How? (2/3)

```
# message informations hash
  my $pg_msg = {
    'sess_hash' => $session_hash, # basically IP+src port
    'timestamp' => $timestamp, # timestamps of the message
    'type' => $type, # one-char type from pg proto
    'data' => $data, # the message data w/o type and length
    ## other fields specifics to each messages are added bellow
  };
```

9 How? (3/3)

pgShark::Core can bind a function to every available messages from the PostgreSQL protocol:

```
Authentication* BackendKeyData Bind BindComplete CancelRequest
Close CloseComplete CommandComplete CopyData CopyDone CopyFail
CopyInResponse CopyOutResponse DataRow Describe
EmptyQueryResponse ErrorResponse Execute Flush NoData
NoticeResponse
NotificationResponse ParameterDescription ParameterStatus Parse
ParseComplete PasswordMessage PortalSuspended Query
ReadyForQuery
RowDescription SSLAnswer+ SSLRequest+ StartupMessage Sync
Terminate
```

10 What?

Some alpha/beta code distributed with pgShark:

- Debug
- SQL / Normalize
- Fouine

11 TODO

- split the project:
  - pgShark::Core
  - tools using pgShark
- next tool base on pgShark: Replay?
- bugfixes

12 Where ?

- https://github.com/dalibo/pgshark
- Contact: jgdr@dalibo.com