pgAudit Overview

David Steele
Crunchy Data

Crunchy Storm
November 28, 2018

About David

- Principal Architect at Crunchy Data, the Trusted Open Source Enterprise PostgreSQL Leader.
- Actively developing with PostgreSQL since 1999.
- PostgreSQL Contributor.
- Primary author of pgBackRest and co-author of pgAudit.

CRUNCHY
Enterprise PostgreSQL

Why pgAudit?

- The goal of the PostgreSQL Audit extension (pgAudit) is to provide PostgreSQL users with capability to produce audit logs often required to comply with government, financial, or ISO certifications.
- Organizations may also have internal requirements that can be satisfied with pgAudit.
- Can also be used for detailed debugging, metrics, and monitoring.

CRUNCHY Enterprise PostgreSQL

Example (log_statement = all)

- User statement:

```
DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;
```

- What gets logged:

```
LOG: statement: DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;
```


CRUNCHY Enterprise PostgreSQL

Example (log_statement = all)

- User statement:

```
DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;
```

- What gets logged:

```
LOG: statement: DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;
```


CRUNCHY
Enterprise PostgreSQL

Example (log_statement = all)

- User statement:

```
DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;
```

- What gets logged:

```
LOG: statement: DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;
```


CRUNCHY
Enterprise PostgreSQL

Example (pgAudit)

- User statement:

```
DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;
```

- What gets logged:

```
AUDIT: SESSION,33,1,FUNCTION,DO,,,"DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;"
AUDIT: SESSION,33,2,DDL,CREATE TABLE,TABLE,public.important_table,CREATE TABLE important_table (id INT)
```


CRUNCHY Enterprise PostgreSQL

Example (pgAudit)

- User statement:

```
DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;
```

- What gets logged:

```
AUDIT: SESSION,33,1,FUNCTION,DO,,,"DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;"
AUDIT: SESSION,33,2,DDL,CREATE TABLE,TABLE,public.important_table,CREATE TABLE important_table (id INT)
```


CRUNCHY Enterprise PostgreSQL

Example (pgAudit)

- User statement:

```
DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;
```

- What gets logged:

```
AUDIT: SESSION,33,1,FUNCTION,DO,,,"DO $$
BEGIN
   EXECUTE 'CREATE TABLE import' || 'ant_table (id INT)';
END $$;"
AUDIT: SESSION,33,2,DDL,CREATE TABLE,TABLE,public.important_table,CREATE TABLE important_table (id INT)
```


CRUNCHY Enterprise PostgreSQL

Stability

- Stability is one of the primary goals of pgAudit.
- Each release is maintained on a separate branch like PostgreSQL (e.g. REL_11_STABLE).
- Only bug fixes are back-patched.

CRUNCHY
Enterprise PostgreSQL

The Future

- Add new logging class MISC_SET.
- Add user settable comment field.

CRUNCHY
Enterprise PostgreSQL

Questions?

website: `http://www.pgaudit.org`

email: david@crunchydata.com

source: `https://github.com/pgaudit/pgaudit`

CRUNCHY
Enterprise PostgreSQL