

## Untrusted

`AUTH` Authenticates Client

```
pub fn auth(&mut self) -> Result<(), std::io::Error>
```

- Value:
    - `AUTHKEY`
  - Returns:
    - `UNTRUSTED_MODE`
    - `ALREADY_AUTHENTICATED`
    - `AUTH_OK`
    - `INV_AUTH_KEY`
- 

## Trusted

`SET_MACHINE_TYPE`

```
pub fn set_type(&mut self) -> Result<(), std::io::Error>
```

- Value:
    - `CONTROLLER`
    - `BUILD`
  - Returns:
    - `CMD_OK`
    - `INV_MACHINE_TYPE`
    - `AUTH_REQUIRED`
- 

`SET_MACHINE_NAME`

```
pub fn set_name(&mut self) -> Result<(), std::io::Error>
```

- Value:

- NAME

- Returns:

- CMD\_OK
- 

#### CHECKOUT\_PACKAGE

```
pub fn checkout(&mut self, pkg_name: &str) -> Result<(), std::io::Error>
```

- Value:

- PKG\_NAME

- Returns:

- INV\_PKG\_NAME
  - {JSON:PKGBUILD}
- 

#### SUBMIT\_PACKAGE

```
pub fn submit(&mut self, path: &str) -> Result<(), std::io::Error>
```

- Value:

- {JSON:PKGBUILD}

- Returns:

- INV\_PKG\_BUILD
  - CMD\_OK
-

## RELEASE\_BUILD

```
pub fn build(&mut self, rb: bool, pkg_name: &str) -> Result<(), std::io::Error>
```

- Value:

- PKG\_NAME

- Returns:

- RELEASE\_ENV\_UNAVAILABLE
  - INV\_PKG\_NAME
  - BUILD\_REQ\_SUBMIT\_IMMEDIATELY
  - BUILD\_REQ\_QUEUED
  - PKG\_BUILD\_DAMAGED
- 

## CROSS\_BUILD

```
pub fn build(&mut self, rb: bool, pkg_name: &str) -> Result<(), std::io::Error>
```

- Value:

- PKG\_NAME

- Returns:

- CROSS\_ENV\_UNAVAILABLE
  - INV\_PKG\_NAME
  - BUILD\_REQ\_SUBMIT\_IMMEDIATELY
  - BUILD\_REQ\_QUEUED
  - PKG\_BUILD\_DAMAGED
- 

## VIEW\_LOG

- Value:
    - `JOB_ID`
  - Returns:
    - `INV_JOB_ID`
    - `NO_LOG`
    - `{JSON:LOG}`
- 

#### `VIEW_SYS_EVENTS`

- Value:
  - Returns:
    - `{JSON:EVENTS}`
- 

#### `GET_DEPENDERS`

```
pub fn get_dependers(&mut self, pkg_name: &str) -> Result<(), std::io::Error>
```

- Value:
    - `PKG_NAME`
  - Returns:
    - `INV_PKG_NAME`
    - `{JSON:DEPENDERS}`
- 

#### `REBUILD_DEPENDERS`

```
pub fn rebuild_dependers(&mut self, pkg_name: &str) -> Result<(),  
std::io::Error>
```

- Value:

- `PKG_NAME`

- Returns:

- `INV_PKG_NAME`
  - `RELEASE_ENV_UNAVAILABLE`
  - `CROSS_ENV_UNAVAILABLE`
  - `CMD_OK`
- 

#### `COMPLETED_JOBS_STATUS`

- Value:

- Returns:

- `{JSON:[JOB]}`
- 

#### `RUNNING_JOBS_STATUS`

- Value:

- Returns:

- `{JSON:[JOB]}`
- 

#### `QUEUED_JOBS_STATUS`

- Value:

- Returns:

- `{JSON:[JOB]}`
- 

#### `CONNECTED_CONTROLLERS`

- Value:

- Returns:

- `{JSON:CONNECTED_CONTROLLERS}`
- 

#### CONNECTED\_BUILDBOTS

- Value:
  - Returns:
    - `{JSON:CONNECTED_BUILDBOTS}`
- 

#### MANAGED\_PACKAGES

```
pub fn get_packages(&mut self) -> Result<(), std::io::Error>
```

- Value:
  - Returns:
    - `{JSON:MANAGED_PACKAGES}`
- 

#### MANAGED\_PKGBUILDS

```
pub fn get_packagebuilds(&mut self) -> Result<(), std::io::Error>
```

- Value:
  - Returns:
    - `{JSON:MANAGED_PKGBUILDS}`
- 

#### CLEAR\_COMPLETED\_JOBS

```
pub fn clear_completed(&mut self) -> Result<(), std::io::Error>
```

- Value:

- Returns:
    - JOBS\_CLEARED
- 

#### CANCEL\_QUEUED\_JOB

```
pub fn cancel_job(&mut self, job_id: &str) -> Result<(), std::io::Error>
```

- Value:
    - JOB\_ID
  - Returns:
    - INV\_JOB\_ID
    - JOB\_CANCELED
- 

#### CANCEL\_ALL\_QUEUED\_JOBS

```
pub fn cancel_all_jobs(&mut self) -> Result<(), std::io::Error>
```

- Value:
  - Returns:
    - JOBS\_CANCELED
- 

#### SUBMIT\_SOLUTION\_RB

```
pub fn submit_sol(&mut self, rb: bool, path: &str) -> Result<(),  
std::io::Error>
```

- Value:
  - {SOLUTION}
- Returns:

- `RELEASE_ENV_UNAVAILABLE`
  - `INV_SOL`
  - `PKG_BUILD_MISSING {NAME}`
  - `BATCH_QUEUED`
- 

#### `SUBMIT_SOLUTION_CB`

```
pub fn submit_sol(&mut self, rb: bool, path: &str) -> Result<(),  
std::io::Error>
```

- Value:

- `{SOLUTION}`

- Returns:

- `CROSS_ENV_UNAVAILABLE`
  - `INV_SOL`
  - `PKG_BUILD_MISSING {NAME}`
  - `BATCH_QUEUED`
- 

#### `GET_CLIENT_INFO`

- Value:

- `CLIENT_NAME`

- Returns:

- `INV_CLIENT_NAME`
  - `{JSON:CLIENT_INFO}`
- 

#### `GET_LOCKED_PACKAGES`



- Value:
  - Returns:
    - `{JSON:PACKAGES}`
- 

#### DELETE\_PKGBUILD

- Value:
    - `PKG_NAME`
  - Returns:
    - `INV_CMD`
    - `INV_PKG_NAME`
    - `REQUIRED_PKG`
    - `CMD_OK`
- 

#### GET\_MANAGED\_EXTRA\_SOURCES

```
pub fn get_extra_sources(&mut self) -> Result<(), std::io::Error>
```

- Value:
  - Returns:
    - `{JSON:[EXTRA_SOURCE]}`
- 

#### REMOVE\_EXTRA\_SOURCE

- Value:
  - `ES_NAME`
- Returns:
  - `INV_ES_NAME`

- CMD\_OK
- 

#### TRANSFER\_EXTRA\_SOURCE

- Value:
    - {JSON:EXTRA\_SOURCE}
  - Returns:
    - BYTE\_COUNT\_ERR
    - CMD\_OK
- 

#### COMPLETE\_TRANSFER

- Value:
  - Returns:
    - ERR\_COULD\_NOT\_INSERT
    - CMD\_OK
- 

## Buildbot Only

#### SET\_MACHINE\_INFORMATION

- Value:
  - Returns:
- 

#### SIG\_READY

- Value:
  - Returns:
- 

#### PONG

- Value:
  - Returns:
- 

#### GET\_DEPLOYMENT\_CONFIG

- Value:
  - Returns:
- 

#### REPORT\_STATUS\_UPDATE

- Value:
  - Returns:
- 

#### REPORT\_STATUS\_UPDATE

- Value:
  - Returns:
- 

#### SUBMIT\_LOG

- Value:
  - Returns:
- 

#### FILE\_TRANSFER\_MODE

- Value:
  - Returns:
- 

#### REPORT\_SYS\_EVENTS

- Value:

- Returns:
- 

#### EXTRA\_SOURCE\_INFO

- Value:
  - Returns:
- 

#### FETCH\_EXTRA\_SOURCE

- Value:
  - Returns:
- 

## Other

#### !INVALID

- Value:
  - Returns:
- 

## JSON

#### Extra Source

```
struct ExtraSource {  
    id: String,  
    filename: String,  
    description: String,  
}
```

---

#### Job

```
struct Job {  
    build_pkg_name: String,  
    job_status: String,  
    job_id: String,  
    requesting_client: String,  
}
```